A NEW SEQUENTIAL APPROXIMATION METHOD

By

DOUGLAS FARIES

Bachelor of Science
Western Illinois University
Macomb, Illinois
1984

Master of Science
Western Illinois University
Macomb, Illinois
1986

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
DOCTOR OF PHILOSOPHY
MAY, 1990

A NEW SEQUENTIAL APPROXIMATION METHOD

Thesis Approved:

Barry Kurt Moser

Thesis Advisor

J. Leroy Folks

David L. Weeks

Kenneth E Case

Norman N. Durham

Dean of Graduate College

ACKNOWLEDGMENTS

I would like to thank all those who have provided the support that I needed during my years at Oklahoma State. First of all, I would like to thank my advisor, Dr. Barry Moser, for giving me the opportunity to join him in researching sequential approximation methods. I truly appreciate the many hours of time he spent discussing the research with me. I feel fortunate to have an advisor who was enthusiastic about the research we were doing. His guidance, timely suggestions, and insights were priceless.

I would also like to thank my advisory committee. To Dr. Leroy Folks, the committee chairman, a special thanks for his questions that made me look at a problem from more than one direction. Also, his support in obtaining scholarships, summer internships, and assistantships was invaluable. Thank you to Dr. David Weeks, for his thought provoking questions and the confidence he gave me. To Dr. Ken Case, thanks for taking time out of his busy schedule. His Quality Control classes were both challenging and fun. Also, thanks to Dr. Ron McNew, who accepted a position at Arkansas before I completed my work. His comments on the paper that Dr. Moser and I submitted for publication were extremely helpful. All of the faculty and graduate students

I encountered were more than willing to share their knowledge and help in any way possible. They made Oklahoma State an enjoyable place to study.

Finally, I would like to thank my parents for their financial support in starting college and for instilling in me an appreciation of academics. I always knew that they were behind me. Also, to my wife Karen, for putting up with some long hours of work. Her love and prayers were exactly what I needed to keep me going.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION AND LITERATURE REVIEW

Let $Y(x)$ be a random variable with distribution function $P(Y(x) \leq y) = F(y,\theta|x)$ and expectation of $Y(x)$ given by

$$M(x) = \int_{-\infty}^{\infty} y \; \partial F(y|x). \tag{1}$$

Assume that neither the distribution function nor the expectation of the random variable is known. Consider the problem of sequentially selecting the values of x, the design levels, in order to efficiently estimate the entire function $M(x)$. From the estimate of $M(x)$, any root, L_p, of the equation $M(x) = p$ can then be estimated. Figure 1 on page 2 depicts the situation graphically.

Sequential approximation methods have beeen applied in several areas of research. One area in which sequential approximation methods are frequently applied is sensitivity analysis. In sensitivity tests, each specimen is assumed to have a critical threshold. The specimen will respond only if a stress greater than its critical threshold is applied. Dixon and Mood (1948) and Neyer (1989) present examples of sequential approximation methods for use in explosives research.

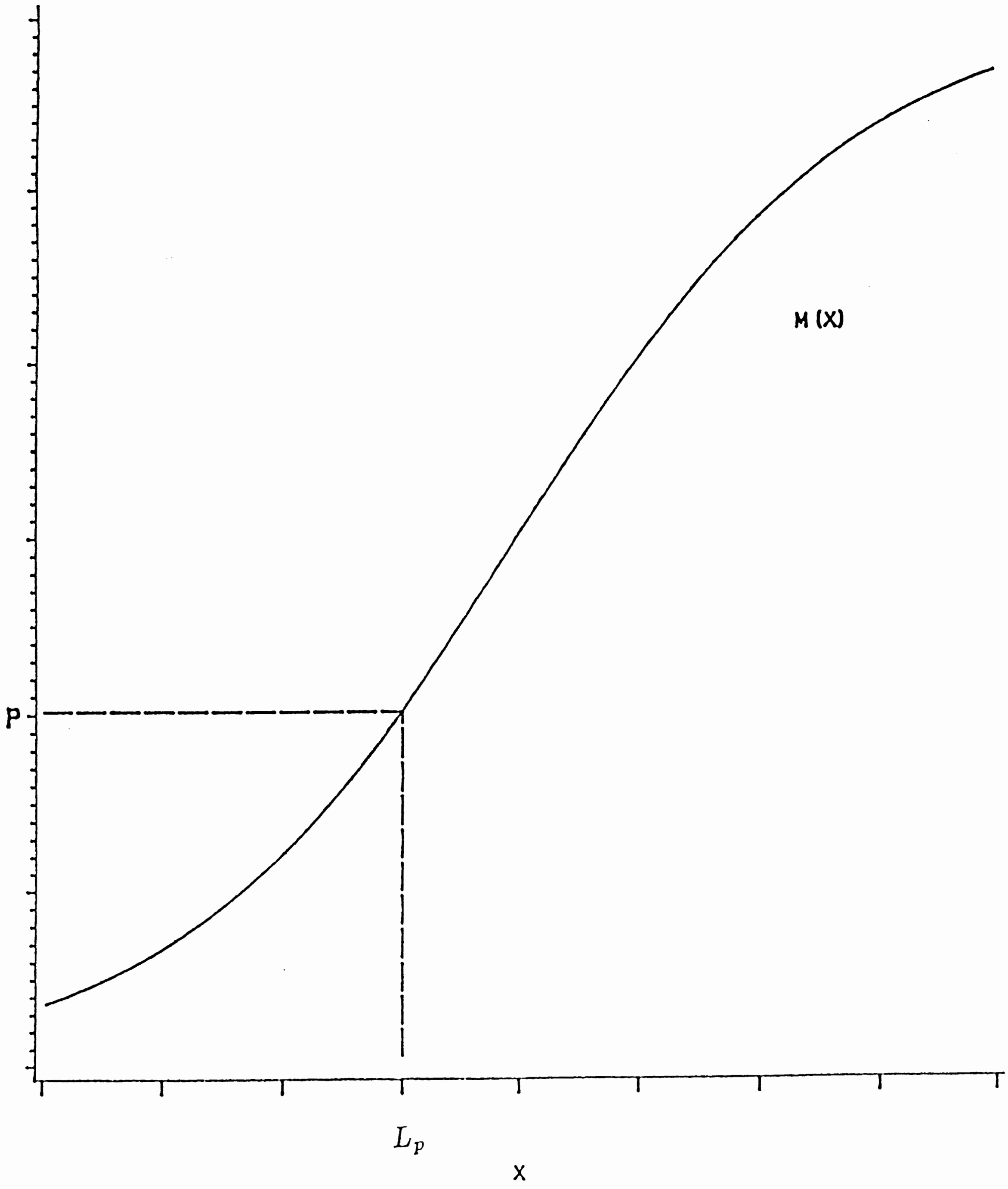

Figure 1. Sequential Approximation

Cochran and Davis (1964) and Davis (1971) study sequential approximation methods for estimating the LD50 in bioassays. These authors demonstrated the superiority of sequential methods to traditional nonsequential designs. In their simulation study with a fixed sample size, the sequential methods produced lower mean square errors than the nonsequential designs. When estimating the LD50 with a specified standard error, they noted an important advantage of the sequential methods. Fewer animals would be required for an experiment using the sequential methods than with the nonsequential designs. Other areas of application include entomology, reliability and educational testing.

Other authors have considered the problem of estimating a single root of the function M(x). As previously mentioned, Dixon and Mood (1948) developed the Up and Down method for use in sensitivity analysis. For example, the sensitivity of explosive material can be defined as the impact needed to detonate the material. Weights are dropped from various heights onto samples of the explosive material and the response, explosion or no explosion, is recorded. The Up and Down method requires a starting height, X_1, and a step size, Δ. Based on the previous responses, the procedure successively generates the height to drop the weight for the next experiment. Successive observations (Y_n = 0 represents "no explosion" and Y_n = 1 represents an "explosion") are taken at heights X_n, determined by the rule

$$X_{n+1} = X_n + \Delta \quad \text{if } Y_n = 0 \text{ and} \tag{2}$$

$$X_{n+1} = X_n - \Delta \quad \text{if } Y_n = 1 ,$$

where Δ is the predetermined positive step size.

Label the heights used in the experiment, from lowest to highest, by h = 0, 1, 2, etc.. Let B denote the lowest height used in the experiment (the height corresponding to h=0). Let n_h be the number of explosions at height level h, and let N be the total number of explosions. The estimate of $L_{.5}$, the height at which 50 percent of the samples will detonate, is given by

$$\hat{L}_{.5} = B + \Delta\cdot[(1/N) \sum hn_h - (1/2)]. \tag{3}$$

Wetherill (1963) noted the effectiveness of the procedure for small or medium sample sizes is highly dependent upon the choice of starting value X_1 and step size Δ.

Robbins and Monro (1951) suggested estimating the single root L_p with the updating rule

$$X_{n+1} = X_n - A_n(Y_n - p) , \tag{4}$$

where y_n is the response associated with X_n, p is a single predetermined constant, and A_n is a fixed sequence of positive constants. Thus, the step size from X_n to X_{n+1} is not a single fixed constant, as it is in the Up and Down method. The term X_{n+1} serves as the estimator of L_p after n updates. Under the following conditions on M(x) and $\{A_n\}$, Robbins and Monro demonstrated that X_n converges to L_p in L^2.

a) M(x) is a nondecreasing function; $M(L_p) = p$; $M'(L_p) > 0$.

b) $\exists$ a positive constant C such that $P[|Y(x)| \leq C] =$

$$\int_{-c}^{c} \partial H(y|x) = 1 \quad \text{for all } x.$$

c) A_n is a sequence of type $1/n$ (There exists positive constants, d_1 and d_2, such that $[d_1/n] \leq A_n \leq [d_2/n]$).

Blum (1954) and Goodsell and Hanson (1976) provided conditions for which X_n converges to L_p almost surely. Using $\{A_n\} = A / n$, for some positive constant A, Chung (1954) and Sacks (1958) defined criteria under which $\sqrt{n}(X_n - L_p)$ is asymptotically normal with mean L_p and variance,

$$\text{var}\{\sqrt{n}(X_n - L_p)\} \;=\; A^2 \cdot \sigma^2 \,/\, (2 \cdot A \cdot M'(L_p) - 1), \tag{5}$$

where $\sigma^2 = \text{var}(Y|x{=}L_p)$ and $M'(L_p) = \left(\partial M(x)/\partial x\right)\Big|_{x=L_p}$. Chung (1954) also showed that the asymptotic variance of X_n is minimized when $A_n = (n \cdot M'(L_p))^{-1}$. Thus, a RM procedure with $A_n = (n \cdot M'(L_p))^{-1}$ represents an optimal RM process. Note that $M'(L_p)$ is the slope of the tangent line of the expectation curve at the root L_p.

In practice $M'(L_p)$ is usually unknown. Thus, an educated guess of $M'(L_p)$ must be made prior to the experiment. Wetherill (1963) demonstrated that a poor guess of $M'(L_p)$ and the starting value X_1 will make the RM procedure (4) inefficient for small and medium sample sizes.

Venter (1967) and Anbar (1977) showed that the desirable properties of the RM procedure are maintained if $M'(L_p)$ is replaced by a strongly consistent estimator. Venter's procedure involves taking two observations, Y and Y', at $X_n - W_n$ and $X_n + W_n$, where $\{W_n\}$ is a sequence of constants converging to zero. Anbar (1977) suggested estimating $M'(L_p)$ by

$$b_n = \sum_{m}^{n-1} y_i(x_i - \bar{x}_{n-1}) \Big/ \sum_{m}^{n-1} (x_i - \bar{x}_{n-1})^2, \tag{6}$$

the slope of the least squares line through $(x_{m(n)}, y(x_{m(n)})), \ldots, (x_{n-1}, y(x_{n-1}))$ for some $m(n) < n$. The design levels are generated by the rule

$$X_{n+1} = X_n - (n \cdot b_n)^{-1}(Y_n - p) \ . \tag{7}$$

Anbar then proved $b_n \xrightarrow{a.s.} M'(L_p)$, $X_n \xrightarrow{a.s.} L_p$, and that $\sqrt{n}(X_n - L_p)$ has the same asymptotic distribution as the optimal RM process.

Wu (1985) suggested estimating the root L_p from an estimate of the entire function M(x). He noted that a smooth nonparametric estimate of M(x) was not feasible without a large number of observations. Therefore, to produce his estimators, he used a parametric form, $H(x|\theta)$, $\theta = (\theta_1, \ldots, \theta_k)'$ for the expectation of Y. Wu proposed the following updating rule:

1) Find an efficient estimate $\hat{\theta}^{(n)}$ for θ based on the n observations $[(y_i, x_i)_1^n]$. (8)
2) Define the estimated expectation function $\hat{H}_n(x) = H(x|\hat{\theta}^{(n)})$ and choose x_{n+1} such that $\hat{H}_n(x_{n+1}) = p$. After n updates, x_{n+1} provides an estimate of L_p.

He suggested using the maximum likelihood estimator, $\hat{\theta}^{(n)}$, as the efficient estimator of θ at each update. Using a one parameter logit expectation, for a binary random variable Y, Wu demonstrated that his procedure is equivalent to a Robbins-Monro process. Thus, X_n from (8) converges to L_p

almost surely, regardless of the true distribution of Y. Using a two parameter logit expectation, Wu was unable to prove consistency. However, assuming consistency, he showed that a first order approximation to his procedure is asymptotically equivalent to the optimal RM procedure.

These procedures are sequential since they generate the n^{th} design level based on the previous $n - 1$ design levels and responses. One approach, however, is to select a fixed total number of observations for the experiment. If a sequential stopping rule is desired, then one can be constructed based on the asymptotic variance of the estimator. Freedman (1970) introduced stopping rules for the Up and Down method based on Bayesian decision theory. The case of a fixed total number of observations is considered in this paper.

Note that procedures (4), (7), and (8) were developed to estimate a single root of the expectation M(x). With the exception of Wu's procedure, no attempt was made to estimate the entire function M(x). Even though M(x) was estimated at each stage in Wu's procedure, the purpose was to provide an estimate of a single L_p, not the entire curve M(x).

In Chapter II of this thesis, a new sequential approximation method called SAM is proposed. The objective of the new procedure is to provide estimates of any number of roots, L_p, of $M(x) = p$. In Chapter III, SAM is studied in detail for binary data applications. Conditions under which SAM's estimates are consistent are provided. Using the two parameter logit expectation, SAM is shown to be asymptotically equivalent (in first order) to a two dimensional RM process . The results of a simulation study are presented in Chapter IV.

CHAPTER II

A NEW SEQUENTIAL APPROXIMATION METHOD (SAM)

The Procedure

As in Wu's procedure (8), to produce root estimators, a parametric model, $G(x|\theta) = \int yg(y|x,\theta)\, \partial y$, is used for the expectation function of Y. Select k unique constants $p_1,\ldots,p_k$, where k is the dimension of the vector of parameters, $\theta = (\theta_1,\ldots,\theta_k)'$. The updating rule for SAM is:

(9)

1) Calculate the MLE of θ using $g(y|x,\theta)$ as the density function of Y; $\hat{\theta}^{(n)} = \hat{\theta}[(y_{ij},x_{ij})]_{(1,1)}^{(n,k)}$ where $(i,j) = (1,1)$ to (n,k) refers to the k observations that that are taken at each of the n updates of the process.

2) Define the estimated expectation after the n^{th} update of the process as

$$\hat{G}_n(x) = G(x|\hat{\theta}^{(n)}) ,$$

and choose the next k dimensional design point $(x_{n+1,1}, \ldots, x_{n+1,k})$ such that $\hat{G}(x_{n+1,j}) = p_j$ for $j = 1,\ldots,k$.

After n updates, $x_{n+1,j}$ provides an estimate of L_{p_j}, for $j = 1,\ldots, k$. In general, estimates of any root, L_{p*}, of the equation $M(x) = p^*$, are provided by $\hat{L}^{(n)}_{p*}$, where $\hat{G}_n(\hat{L}^{(n)}_{p*}) = p^*$.

Note that both the design levels, $x_{i,j}$, and the responses, $y_{i,j}$, are random variables. However, the joint probability density function, $g(x_{1,1},y_{1,1},\ldots,x_{n,k},y_{n,k})$, is simply the product of the conditional probability densities of $Y_{i,j}$, given $x_{i,j}$. That is,

$$g(x_{1,1},y_{1,1},\ldots,x_{n,k},y_{n,k}) = \prod_{i=1}^{n} \prod_{j=1}^{k} g(y_{i,j}|x_{i,j}). \tag{10}$$

To see this, note

$$\begin{aligned} g(x_{1,1},y_{1,1},\ldots,x_{n,k},y_{n,k}) = \\ g(y_{n,1},\ldots,y_{n,k}|x_{1,1},y_{1,1},\ldots,y_{n-1,k},x_{n,1},\ldots,x_{n,k})\cdot \\ P(x_{n,1},\ldots,x_{n,k}|x_{1,1},y_{1,1},\ldots,x_{n-1,k},y_{n-1,k})\cdot \;\ldots \\ g(y_{1,1},\ldots,y_{1,k}|x_{1,1},\ldots,x_{1,k})\cdot P(x_{1,1},\ldots,x_{1,k}). \end{aligned} \tag{11}$$

The random variables $Y_{n,1},\ldots,Y_{n,k}$, given the design levels $x_{n,1},\ldots,x_{n,k}$, are assumed to be independent random variables. Also, $g(y_{i,j}|x_{1,1},y_{1.1},\ldots,x_{i-1,k},y_{i-1,k},$ $x_{i,1},\ldots,x_{i,k}) = g(y_{i,j}|x_{i,j})$. Since $P(x_{i,1},\ldots,x_{i,k}|x_{1,1},y_{1,1},\ldots,x_{i-1,k},y_{i-1,k}) = 1$ and $x_{1,1},\ldots,x_{1,k}$ are fixed values, (11) simplifies to the result in (10). Therefore, in the maximum likelihood calculations, the likelihood function is $\prod_{i}^{n} \prod_{j}^{k} g(y_{i,j}|x_{i,j},\theta)$, considered as a function of θ.

SAM uses MLEs to calculate the next design levels. However, when only a few observations have been taken, MLEs may not exist. Therefore, some other procedure is needed to produce the initial design levels until the MLEs exist. Two methods for producing the initial design levels are given in the simulation study of Chapter IV. In the first, an initial set of five design levels is chosen. A total of ten responses are observed at these levels. The second method starts with a Robbins-Monro procedure and switches to SAM's updating rules when the MLEs first exist. Two other possibilities for producing the initial design levels are the modified binary search presented by Neyer (1989) and the two dimensional Robbins-Monro process proposed by Moser and Fei (1989a).

For $k = 1$, SAM (9) is equivalent to the MLE version of Wu's procedure (8), provided $H(x|\theta) = G(x|\theta)$. However, for $k > 1$, the two procedures differ. At the n^{th} update, SAM generates k new design levels , $x_{n+1,1}$, ..., $x_{n+1,k}$, while Wu's procedure generates a single level, x_{n+1}.

Figure 2 on the following page is a graphical display of SAM's updating rule. A sketch of the estimated expectation curve after n updates, $\hat{G}_n(x)$, is given. The dotted lines highlight the next k design levels ($x_{n+1,1},\ldots,x_{n+1,k}$), the solutions to $\hat{G}_n(x) = p_j$, $j = 1,\ldots,k$.

The difference between SAM and Wu's procedure is also apparent in Figure 2. The design levels in Wu's procedure will be grouped around a single point, L_p. Using SAM, the design levels will be in k separate groups, around L_{p_1}, L_{p_2}, ..., and L_{p_k}.

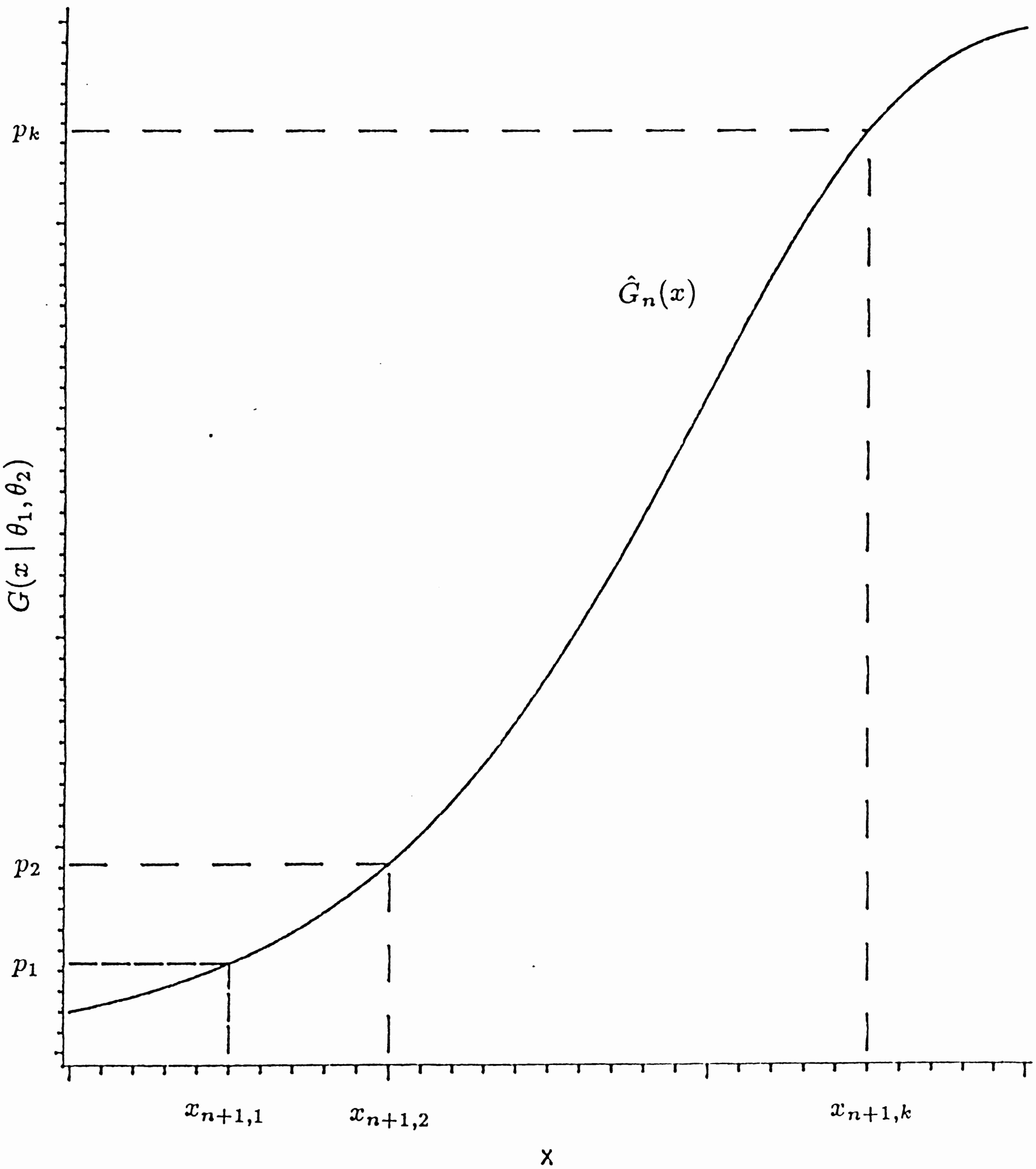

Figure 2. SAM

Selecting $p_1, \ldots, p_k$

SAM's updating rule requires the choice of k unique constants p_1, p_2, ..., p_k. One approach is to base the selection of p_1, p_2, ..., p_k on $\mathrm{Var}(\hat{L}^{(n)}_{p*})$. This variance is a function of M(x). Since M(x) is unknown, a different criteria must be used to select p_1, p_2, ..., p_k.

For a given set of values $p_1, p_2, \ldots, p_k$, and for any $p^* \in \Omega$, where Ω is the range space of $G(x|\theta)$, let

$$\sigma_n^{*2} = \lim_{n \longrightarrow \infty} E\left[\hat{L}^{(n)}_{p*} - L_{p*}\right]^2, \tag{12}$$

where the expectation is calculated with respect to the density function $g(y|x,\theta)$. The value σ_n^{*2} is a function of p^*, $p_1, \ldots, p_k$, denoted by $\sigma_n^{*2}(p^*, p_1, \ldots, p_k)$. The selection of p_1, p_2, ..., p_k will be based on the function $\sigma_n^{*2}(p^*, p_1, \ldots, p_k)$. If $\hat{L}^{(n)}_{p*}$ is a consistent estimator of L_{p*} and $g(y|x,\theta)$ is the true density function of Y, then σ_n^{*2} represents the asymptotic variance of $\hat{L}_{p*}$.

Two different criteria for selecting $p_1, p_2, \ldots, p_k$ are now presented. The first approach is to choose the $p_1, p_2, \ldots, p_k$ that minimize

$$\int_{\Omega^*} \sigma_n^{*2}(u, p_1, \ldots, p_k) \cdot \phi(u) \cdot \partial u \ , \tag{13}$$

where $\phi(\cdot)$ is a measure on $p^* \in \Omega^* \subseteq \Omega$. The space Ω^* can be interpreted as the range of interest for p^*, with $\phi(\cdot)$ indicating the level of interest in any L_{p*}. If all roots, L_{p*}, $p^* \in \Omega^*$, are of equal interest, then $\phi(\cdot)$ is a

Uniform(Ω^*) density function. The $p_1, p_2, \ldots, p_k$ that minimize (13) will be referred to as the average minimum $p_1, p_2, \ldots, p_k$ with respect to $\phi(\cdot)$.

A second approach is to choose the set $p_1, \ldots, p_k$ that minimizes the maximum $\sigma_n^{*2}(p^*, p_1, \ldots, p_k)$ for $p^* \in \Omega^*$. That is, select the $p_1, p_2, \ldots, p_k$ that minimize

$$\sup_{p^* \in \Omega^*} \sigma^{*2}(p^*, p_1, \ldots, p_k) \; . \tag{14}$$

The $p_1, \ldots, p_k$ that minimize (14) will be referred to as the minimax $p_1, \ldots, p_k$. In Chapter III, the average minimum and minimax values of p_1, p_2 are derived when $G(x|\theta)$ is the two parameter logit model.

Bounded and Alternate Versions of SAM

For small sample sizes, estimates of θ from SAM (9) or WU (8), and estimates of $M'(L_p)$ from Anbar (6) are extremely variable. Thus, the changes in the design levels from the n^{th} to the $(n+1)^{st}$ update can be extremely large. Wu (1985) has shown that bounded versions of these procedures, which limit the step size from X_n to X_{n+1}, improve their performance for small to medium sample sizes. The following is a bounded version of SAM. Let $d_{n,j}$, $j = 1, \ldots, k$, be the solution to $X_{n+1,j} = X_{n,j} - (d_{n,j} / n) \cdot (Y_{n,j} - p_j)$, where X_{n+1} is the solution of $\hat{G}_n(x) = p_j$. The $(n+1,j)^{th}$ design point is then defined by

$$X_{n+1,j} = X_{n,j} - (d^*_{n,j} / n)\cdot(Y_{n,j} - p_j), \tag{15}$$

where $d^*_{n,j} = \max[\delta_1, \min(d_{n,j}, \delta_2)]$ and $\delta_1 < \delta_2$.

Instead of using the solutions of $\hat{G}_n(x) = p_j$ as the next design levels, the bounded version first checks the step sizes from $X_{n,j}$ to $X_{n+1,j}$. If the step sizes are not within the bounds determined by δ_1, δ_2 and n, then the step size bound is used to calculate $X_{n+1,j}$. Note that using $\delta_1 = -\infty$ and $\delta_2 = \infty$ is equivalent to the unbounded version of SAM (8).

Using the bounded version of SAM (15), the step size, $|X_{n+1,j} - X_{n,j}|$, is bounded above by $|(\delta_2 / n)\cdot(Y_{nj} - p_j)|$. Note that as n increases, the maximum allowable step size decreases. If Y is a binary random variable with p = .2, k = 2 and $\delta_2 = 50$, at the 10^{th} update the maximum step size is $|5\cdot(Y_{10,j} - .2)|$. This becomes 1 if $Y_{10,j} = 0$, and 4 if $Y_{10,j} = 1$. At the 30^{th} update, the step size is bounded by $|1.6\cdot(Y_{30,j} - .2)|$. This bound is .32 is $Y_{30,j} = 0$ and 1.28 if $Y_{30,j} = 1$.

Instead of observing k responses at each update, the following adaptation to SAM may be used. At the $(i+1)^{st}$ update, choose $X_{i+1,1}$ to be the solution of $G(x|\hat{\theta}^{(i,1)}) = p_1$, where $\hat{\theta}^{(i,1)}$ is the MLE of θ based upon the previous i updates ($\hat{\theta}^{(i)}$ in the previous notation). After observing the response $Y_{i+1,1}$ at level $X_{i+1,1}$, recalculate $\hat{\theta}$ using $(y_{1,1},x_{1,1}),\ldots,(y_{i,k},x_{i,k})$ and $(y_{i+1,1},x_{i+1,1})$. Denote this estimator by $\hat{\theta}^{(i,2)}$. Choose $X_{i+1,2}$ to be the solution to $G(x|\hat{\theta}^{(i,2)}) = p_2$. Continue this process to produce all k design levels of the $(i+1)^{st}$ update.

The difference between this version of SAM and (9) is that updated parameter estimates are calculated after each observation, instead of after every k observations. This version has the advantage of using all information to calculate each new design level. However, it is often easier to run an experiment with fewer updates, observing several responses at a time. Thus, for the remainder of this paper, the original version of SAM (9) will be used. The asymptotic results in Chapter III follow for either version of SAM.

CHAPTER III

SAM FOR BINARY DATA

Introduction

In Chapter II, a new sequential approximation method, SAM, was proposed. In this Chapter, SAM will be studied when Y is a binary random variable. If Y is a binary random variable, then $M(x) = P(Y = 1|x)$ and L_p is the p^{th} percentile of $M(x)$. To use SAM's updating rule, a parametric model, $G(x|\theta)$, must be selected. Wu (1985) suggested using the logit model for binary data when $M(x)$ is unknown. Thus, without prior knowledge of $M(x)$, $G(x|\theta)$ is chosen to be the two parameter logit model. When $G(x|\theta)$ is the two parameter logit model, SAM (9) is referred to as the logit version of SAM.

The consistency of SAM's estimates is discussed in the first section of this chapter. Three theorems are presented giving conditions for the consistency of SAM's estimators. An example of SAM's updating rule, using the two parameter logit model, is presented next. The logit version of SAM is then shown to be asymptotically equivalent (in first order) to a two dimensional RM procedure. Two approaches for selecting the constants p_1 and p_2 used in the logit version of SAM are also presented. The chapter concludes with a discussion of the asymptotic variances and biases of estimators from the RM procedure and the logit version of SAM.

Consistency

Using a one parameter logit expectation for $H(x|\theta)$, Wu (1985) demonstrated that $\hat{L}_p$ from his procedure (8) is a consistent estimator of L_p. SAM and Wu's procedure are the same when using one parameter expectations, provided $H(x|\theta) = G(x|\theta)$. Thus $\hat{L}_p$ from SAM, using the one parameter logit expectation, is also a consistent estimator of L_p. This result holds regardless of the true expectation, $M(x)$. Using a model other than the one parameter logit for $H(x|\theta)$, Wu was unable to prove consistency.

It is known that under the standard regularity conditions with independent observations, MLEs are both consistent and asymptotically normal. Due to the dependence of the random variables, Y_{1j}, Y_{2j},..., Y_{nj}, it is difficult to demonstrate the consistency of SAM's or Wu's estimators using a general k parameter expectation. As previously mentioned, when $H(x|\theta)$ is the two parameter logit model, Wu was not able to provide a rigorous proof of the consistency of his estimators. However, using the results of Dubins and Freedman (1965), Wu demonstrated that if $\hat{\theta}^{(n)}$ from (8) converges almost surely to a constant θ^*, then $\hat{L}_p$ from (8) converges to L_p almost surely. Theorem 1 applies the work of Dubins and Freedman (1965) to SAM's estimators.

Theorem 1. Let $x_{1,1}, y_{1,1}, \ldots, x_{n,2}, y_{n,2}$ be a sequence of design levels of binary responses from SAM (9), where $G(x|\theta)$ is the two parameter logit expectation. Assume that the MLEs, $(\hat{\theta}_1^{(n)}, \hat{\theta}_2^{(n)})$, converge almost surely to a constant (θ_1^*, θ_2^*), $\theta_2^* \neq 0$. Also assume that $M(x)$ is a strictly increasing function of x. Then $\hat{L}_{p_1}^{(n)}$ and $\hat{L}_{p_2}^{(n)}$ from SAM (9) converge almost surely to L_{p_1} and L_{p_2}, respectively.

The proof is given in Appendix A. Note that Theorem 1 does not claim that $\hat{L}_{p*}$, $p^* \neq p_1, p_2$, is a consistent estimator of L_{p*}. The estimator of any L_{p*}, $p^* \neq p_1, p_2$, is a function of $G(x|\theta_1, \theta_2)$ (see the paragraph following (9)). If $G(x|\theta) \neq M(x)$, then $\hat{L}_{p_1}$ and $\hat{L}_{p_2}$ are still consistent estimators of L_{p_1} and L_{p_2} respectively, although $\hat{L}_{p*}$ in general ($p^* \neq p_1, p_2$) may not be a consistent estimator of L_{p*}.

As mentioned at the beginning of this section, $\hat{L}_p$ from SAM, when $G(x|\theta)$ is the one parameter logit expectation, is a consistent estimator of L_p. The following theorem extends this consistency result to one parameter binary expectations other than the logit model. It does not require the assumption that the MLEs converge almost surely to a constant. However, it requires the strong assumption that the parametric model used in SAM is the true expectation of Y. That is, $G(x|\theta) = M(x)$.

Theorem 2. Let Y be a binary random variable depending upon the level of another variable x. Let $G(x|\theta) = M(x|\theta) = E(Y|x,\theta)$, where θ is a single unknown parameter. Consider the following conditions:

1) $M(x|\theta)$ is continuous in θ,
2) $\exists\ \delta_1, \delta_2 \in R$ such that $\delta_1 < \partial M/\partial\theta < \delta_2$.

If conditions 1) and 2), along with the standard regularity conditions on the distribution of Y (given in Appendix B), are satisfied, then the estimator $\hat{L}_p$ produced by SAM (9) converges in probability to L_p.

The one parameter logit and probit models are examples of expectations which satisfy these conditions. The proof is based in the results of Crowder (1975), and is given in Appendix B.

By placing certain restrictions on the bounded version of SAM (15), it will now be shown that $\hat{L}_{p_1}, \hat{L}_{p_2}, \ldots, \hat{L}_{p_k}$ converge almost surely to $L_{p_1}, L_{p_2}, \ldots, L_{p_k}$. In (15), replace $d^*_{n,j}$ with $d^*_{n-1,j}$ to produce the updating rule

$$X_{n+1,j} = X_{n,j} - (d^*_{n-1,j} / n)\cdot(Y_{n,j} - p_j), \tag{16}$$

where $d^*_{n-1,j} = \max\{\delta_1, \min(d_{n-1,j}, \delta_2)\}$, $d_{n-1,j}$ is the solution to $X_{n,j} = X_{n-1,j} - [d_{n-1,j}/(n-1)]\cdot(Y_{n-1,j} - p_j)$ and $X_{n,j}$ is the solution to $\hat{G}_{n-1}(x) = p_j$. The difference between (15) and (16) is that the step size factor in (16), $d^*_{n-1,j}$, is based only upon $x_{1,1}, Y_{1,1}, \ldots, x_{n,k}$, instead of

$x_{1,1}, y_{1,1}, \ldots, x_{n,k}, y_{n,k}$.

Theorem 3. Let Y be a binary random variable with a strictly increasing expectation function, M(x). Let $x_{1,1}, y_{1,1}, \ldots x_{n,k}, y_{n,k}$ be a sequence of design levels and responses from (16), with $0 < \delta_1 < \delta_2$. Then $X_{n+1,j} = \hat{L}^{(n)}_{p_j}$ converges to L_{p_j}, $j = 1,2,\ldots,k$, almost surely.

The proof follows from application 2 of Robbins and Siegmund (1971). Two points regarding Theorem 3 should be noted. By using $d^*_{n-1,j}$ in place of $d^*_{n,j}$, (16) fails to use the latest information in calculating the step sizes. Thus, it is not recommended above the updating rule in (15). The use of $d^*_{n-1,j}$ was simply to satisfy the conditions of Robbins and Siegmund (1971). To apply their results, $d^*_{n-1,j}$ in (16) must be a function of $x_{1,1}, y_{1,1}, \ldots, x_{n,j}$ only (not including $y_{n,j}$). This does not say that Theorem 3 does not hold using $d^*_{n,j}$. An extension of their work, however, would be needed for that result.

Secondly, Theorem 3 states that $\hat{L}_{p_j}$ converges to L_{p_j}, for $j = 1,\ldots,k$. Thus we have consistent estimators for k roots, $L_{p_1},\ldots,L_{p_k}$. As in Theorem 1, this does not prove the consistency of $\hat{L}_{p*}$ for $p^* \notin \{p_1,\ldots,p_k\}$. The estimate of any L_{p*} is the solution to $\hat{G}_n(x) = p^*$, which depends upon the selected model, $G(x|\theta)$. For $\hat{L}_{p*}$ to converge to L_{p*} for any p^*, the model used by SAM, $G(x|\theta)$, must be the true expectation of Y, M(x).

In this section, three theorems have been presented. Each provides restrictions under which SAM's estimators are consistent. At this time, a rigorous proof, using less restrictive conditions, of the consistency of SAM's estimates has not been developed. However, these three theorems provide significant progress toward this goal.

An Example: The Two Parameter Logit Model

Let Y be a binary random variable with expectation M(x). Consider using SAM's updating rule (9), with $G(x|\theta)$ given by the two parameter logit model. That is, let

$$G(x|\theta_1,\theta_2) = (1 + \exp\{-\theta_2(x-\theta_1)\})^{-1}. \tag{17}$$

Since the logit model is symmetric, let $p_1 = p$ and $p_2 = 1 - p$ for some $0 < p < 1/2$. At the n^{th} update, SAM (9) consists of observing $y_{n,1}$ and $y_{n,2}$ at the two design levels, $x_{n,1}$ and $x_{n,2}$. The next two design levels, $x_{n+1,1}$ and $x_{n+1,2}$, are then generated by

$$\begin{aligned} x_{n+1,1} &= \hat{\theta}_1^{(n)} - [\ln\{(1-p)/p\} \,/\, \hat{\theta}_2^{(n)}] \quad \text{and} \\ x_{n+1,2} &= \hat{\theta}_1^{(n)} + [\ln\{(1-p)/p\} \,/\, \hat{\theta}_2^{(n)}] \,, \end{aligned} \tag{18}$$

where the maximum likelihood estimates $\hat{\theta}_1^{(n)}$ and $\hat{\theta}_2^{(n)}$ are determined by the normal equations

$$\sum_{i=1}^{n} \sum_{j=1}^{2} y_{ij} = \sum_{i=1}^{n} \sum_{j=1}^{2} (1 + \exp\{-\theta_2(x_{ij} - \theta_1)\})^{-1} \tag{19}$$

$$\sum_{i=1}^{n} \sum_{j=1}^{2} x_{ij} y_{ij} = \sum_{i=1}^{n} \sum_{j=1}^{2} x_{ij} (1 + \exp\{-\theta_2 (x_{ij} - \theta_1)\})^{-1} .$$

For example, consider the following set of ten observations.

Design Level	Response	Design Level	Response
2.0	0	4.0	0
2.0	0	4.5	1
3.0	0	4.75	0
3.0	1	5.0	1
4.0	0	5.0	1

Using the Newton Raphson method to solve the normal equations (19), the MLEs of θ_1 and θ_2 are $\hat{\theta}_1^{(5)} = 4.250$ and $\hat{\theta}_2^{(5)} = 1.144$. By (18), using p = .2, the next design levels are

$$\begin{aligned} x_{6,1} &= \hat{\theta}_1^{(5)} - [\ln\{(1-.2)/.2\} / \hat{\theta}_2^{(5)}] \\ &= 4.250 - [\ 1.386 / 1.144] = 3.038, \end{aligned}$$

and

$$\begin{aligned} x_{6,2} &= \hat{\theta}_1^{(5)} + [\ln\{(1-.2)/.2\} / \hat{\theta}_2^{(5)}] \\ &= 4.250 + [\ 1.386 / 1.144] = 5.462. \end{aligned}$$

Figure 3 on the following page depicts the estimated expectation curve, $\hat{G}_5(x)$, after the initial ten observations. The design levels for the next experiment, 3.038 and 5.462, are emphasized with the dotted lines.

The responses, $y_{6,1}$ and $y_{6,2}$, are then observed when the experiment is run at levels 3.038 and 5.462, respectively. If $y_{6,1} = 1$ and $y_{6,2} = 1$, then $\hat{\theta}_1^{(6)} = 3.836$ and $\hat{\theta}_2^{(6)} = .936$. The design levels of the 7^{th} update are

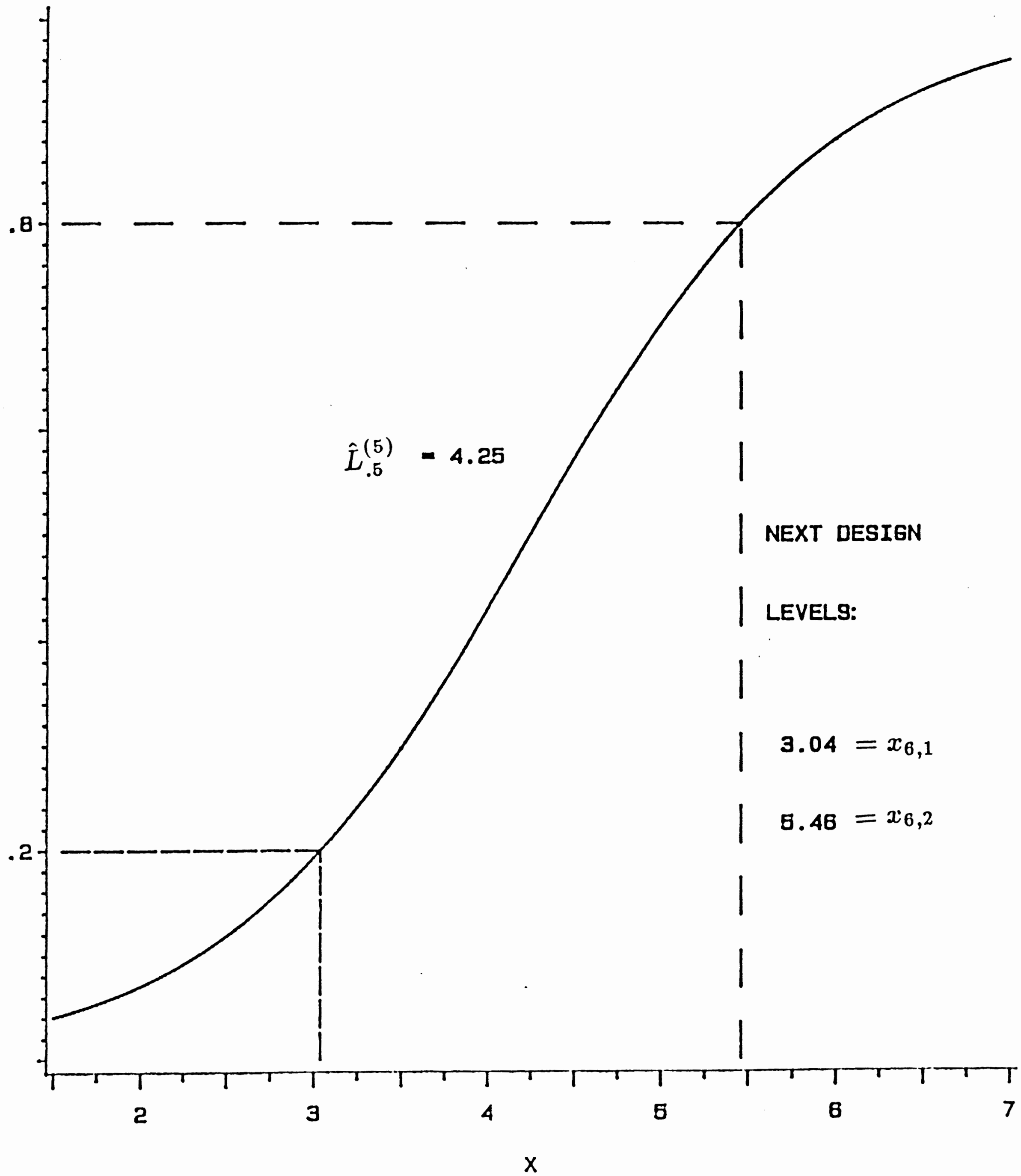

Figure 3. SAM An Example

$x_{7,1} = 2.354$ and $x_{7,2} = 5.318$. Table 1 presents the designs levels and responses through the 12^{th} update.

TABLE 1

AN EXAMPLE

Update	Level 1	Response 1	Level 2	Response 2
1	2.0	0	4.0	0
2	2.0	0	4.5	1
3	3.0	0	4.75	0
4	3.0	1	5.0	1
5	4.0	0	5.0	1
6	3.04	1	5.46	1
7	2.35	0	5.32	1
8	2.62	0	5.06	1
9	2.79	0	4.91	0
10	2.92	0	5.33	1
11	3.04	1	5.20	0
12	2.60	0	5.70	1

$$\hat{\theta}_1^{(12)} = 4.138 \ , \ \hat{\theta}_2^{(12)} = 1.003$$

Using the final estimated expectation function, $\hat{G}_{12}(x) = G(x|\hat{\theta}_1^{(12)}, \hat{\theta}_2^{(12)})$, estimates of the $(p^*)^{th}$ percentile can be constructed for any $p^* \in (0,1)$. From (17),the solution, $\hat{L}_{p*}^{(12)}$, of $\hat{G}_{12}(x) = p^*$ is

$$\hat{L}_{p*}^{(12)} = \hat{\theta}_1^{(12)} - \ln[(1-p^*)/p^*]\cdot(\hat{\theta}_2^{(12)})^{-1}.$$

Thus, the final estimate of the 75^{th} percentile, $L_{.75}$, is

$$\hat{L}_{.75} = 4.138 - \ln[1/3]\cdot(1.003)^{-1} = 5.233.$$

In Appendix E, the codes and descriptions of programs designed to assist a researcher in using SAM's updating rule, with the two parameter logit model, are given.

Equivalence of the Logit Version of SAM and a Two Dimensional RM Procedure

Consider using the two parameter logit model (17) for $G(x|\theta)$ in SAM (9) and for $H(x|\theta)$ in Wu (8). Wu (1985), assuming the consistency of (8), proved that a first order approximation to his procedure is asymptotically equivalent to the optimal RM procedure. Using linear approximations to the two parameter logit model around L_{p_1} and L_{p_2}, a similar result for SAM is now presented.

Since the logit model is symmetric, let $p_1 = p$ and $p_2 = 1-p$ for some $0 < p < 1/2$. Consider the following approximation to $G(x|\theta)$:

near L_p (j=1) (20)

$$(1 + \exp\{-\theta_2(x - \theta_1)\})^{-1} \cong p + (x - L_p)\cdot\lambda_p$$

near L_{1-p} (j=2) (21)

$$(1 + \exp\{-\theta_2(x - \theta_1)\})^{-1} \cong (1 - p) + (x - L_{1-p})\cdot\lambda_{1-p},$$

where $\lambda_p = \lambda_{1-p} = \theta_2 p(1-p)$ are the tangent slopes of $G(x|\theta_1, \theta_2)$ at L_p and L_{1-p} ,respectively. Since $\lambda_p = \lambda_{1-p}$, drop the subscripts and denote both by λ. From Figure 4 on the following page, the approximation is valid for $x_{i,1}$ near L_p and $x_{i,2}$ near L_{1-p}.

Substitute $L_{1-p} = L_p + \{2p(1-p)\cdot\ln[(1-p)/p]\} / \lambda$ into (21) to obtain

near L_{1-p}

$$(1 + \exp(-\theta_2(x - \theta_1)))^{-1} \cong \tag{22}$$
$$(1 - p) + (x - L_p)\cdot\lambda - 2p(1-p)\cdot\ln[(1-p)/p]$$

Applying (20) and (22) to the likelihood equations (19) yields

$$\sum_1^n \{\lambda\cdot(x_{i1} - L_p) + \lambda\cdot(x_{i2} - L_p) +$$
$$1 - 2p(1-p)\cdot\ln[(1-p)/p]\} = \sum_1^n \{y_{i1} + y_{i2}\}$$

and (23)

$$\sum_i^n \{\lambda\cdot(x_{i1}^2 + x_{i2}^2) - \lambda\cdot L_p(x_{i1} + x_{i2}) + px_{i1} +$$
$$(1-p)(1 - 2p\cdot\ln[(1-p)/p])\cdot x_{i2}\} = \sum_i^n \{x_{i1}y_{i1} + x_{i2}y_{i2}\}$$

Estimators of λ and L_p are then obtained by solving (23),

$$\hat{\lambda}_n = \Big(\sum_{i=1}^n \sum_{j=1}^2 (x_{ij} - \bar{x}_n)^2\Big)^{-1}\cdot\Big(\sum_{i=1}^n \sum_{j=1}^2 y_{ij}(x_{ij} - \bar{x}_n) - \tag{24}$$
$$\{(1/2) - p - (1-p)\cdot\ln[(1-p)/p]\}\sum_1^n(x_{i2} - x_{i1})\Big)$$

$$\hat{L}_p^{(n)} = -\big(2n\cdot\hat{\lambda}_n\big)^{-1}\cdot \tag{25}$$
$$\Big(\sum_{i=1}^n \sum_{j=1}^2 \{y_{ij} - x_{ij}\cdot\hat{\lambda}_n\} - n\{1 - 2p(1-p)\cdot\ln[(1-p)/p]\Big)$$

where $\bar{x}_n = \sum_{i=1}^n \sum_{j=1}^2 x_{ij} / 2n$, and the superscript on $\hat{L}_p^{(n)}$ and subscript on $\hat{\lambda}_n$ denote the n^{th} update. Substituting (24)

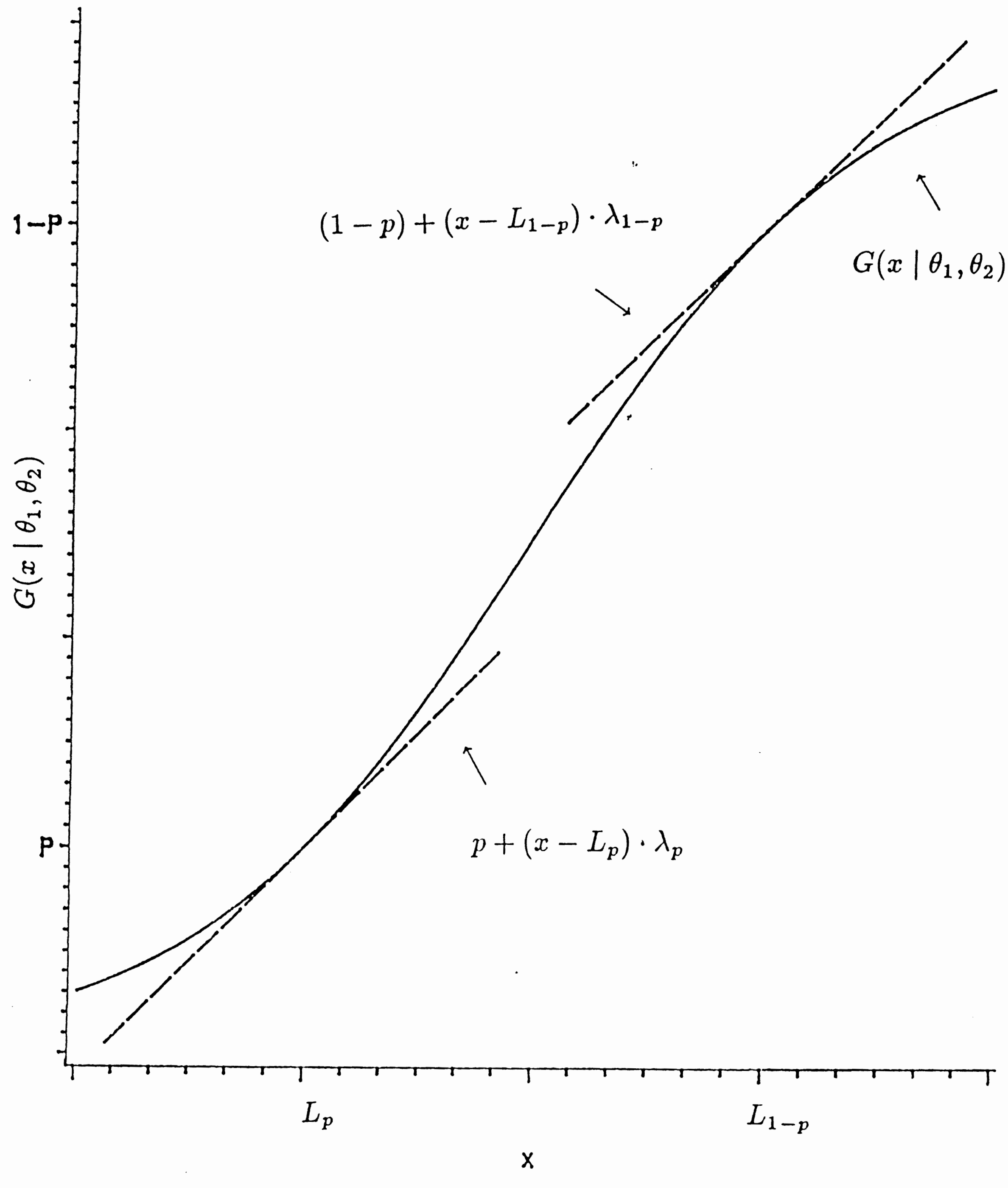

Figure 4. Logit Approximation

into (25) produces

$$\hat{L}_p^{(n)} = \hat{L}_p^{(n-1)} - \left(2n\hat{\lambda}_n\right)^{-1}\Big((y_{n1} - p)\cdot K_1 + (y_{n2} - (1-p))\cdot K_2 - (\hat{\lambda}_n)^{-1}\cdot K_3\Big), \tag{26}$$

where

$$K_1 = \sum_{i=1}^{n}\sum_{j=1}^{2}(x_{ij} - x_{n1})^2 \Big/ \sum_{i=1}^{n}\sum_{j=1}^{2}(x_{ij} - \bar{x}_n)^2,$$

$$K_2 = \left(\sum_{i=1}\sum_{j=1}\{x_{ij} - (1/2)(x_{n1} + x_{n2})\}^2 - ((n-1)/2)(x_{n2} - x_{n1})^2\right) \Big/ \sum_{i=1}\sum_{j=1}(x_{ij} - \bar{x}_n)^2$$

and

$$K_3 = (1/2)(1-2p-2p\cdot\ln[(1-p)/p])\left[1 + \frac{(x_{n1} - \bar{x}_n)\cdot\Sigma(x_{i2} - x_{i1})}{\sum_i\sum_j (x_{ij} - \bar{x}_n)^2}\right].$$

Assuming the consistency of SAM (9), as $n \longrightarrow \infty$, $x_{n1} \longrightarrow L_p$ and $x_{n2} \longrightarrow L_{1-p}$. Therefore, as $n \longrightarrow \infty$, $K_1 \longrightarrow 2$, $K_2 \longrightarrow 0$, and $K_3 \longrightarrow 0$, and from (26),

$$x_{n+1,1} = \hat{L}_p^{(n)} = \hat{L}_p^{(n-1)} - \left(n\cdot\hat{\lambda}_n\right)^{-1}(y_{n1} - p) \tag{27}$$

By similar arguments,

$$x_{n+1,2} = \hat{L}_{1-p}^{(n)} = \hat{L}_{1-p}^{(n-1)} - \left(n\cdot\hat{\lambda}_n\right)^{-1}(y_{n2} - (1-p)) \tag{28}$$

By Theorem 4 in Appendix C, $\hat{\lambda}_n$ converges almost surely to $c_L = \{2p(1-p)\ln[(1-p)/p]/(L_{1-p} - L_p)$. Therefore, (27) and (28) are (first order) asymptotically equivalent to two independent RM procedures, both with $A_n = (n\cdot c_L)^{-1}$. From Sacks (1958), with $\sigma^2 = \lim_{x \to L_p} \mathrm{Var}(Y|x) = \lim_{x \to L_{1-p}} \mathrm{Var}(Y|x)$,

$$\begin{pmatrix} \hat{L}_p^{(n)} \\ \hat{L}_p^{(n)} \end{pmatrix} \sim \text{ASN} \left(\begin{pmatrix} L_p \\ L_p \end{pmatrix}, \begin{bmatrix} c_L^{-2}\sigma^2/n(2c_L^{-1}M'(L_p)-1) & 0 \\ 0 & c_L^{-2}\sigma^2/n(2c_L^{-1}M'(L_{1-p})-1) \end{bmatrix} \right). \quad (29)$$

If $M'(L_p) = M'(L_{1-p}) = c_L$, then (27) and (28) are asymptotically optimal RM procedures. If the true expectation is given by the two parameter logit model, then $M'(L_p) = M'(L_{1-p}) = \theta_2 p(1-p) = c_L$, and (29) becomes

$$\begin{pmatrix} \hat{L}_p^{(n)} \\ \hat{L}_{1-p}^{(n)} \end{pmatrix} \sim \text{ASN} \left(\begin{pmatrix} L_p \\ L_{1-p} \end{pmatrix}, \left[n \cdot \theta_2^2 \cdot p \cdot (1-p)\right]^{-1} \cdot I_2 \right) \quad (30)$$

If $M(x)$ is not the two parameter logit model, then $(nc_L)^{-1}$ may not be optimal. The difference between the logit version of SAM and the optimal RM procedures at p and 1-p can be characterized by the ratios c_L / $M'(L_p)$ and c_L / $M'(L_{1-p})$, respectively. Ratio values of 1 indicate that the logit version of SAM is asymptotically equivalent to two optimal RM processes. The ratio value is a function of the true expectation, $M(x)$, and the value of p. In Table 2, ratios are provided for four models of $M(x)$ (the logit, probit, skewed logit, and loglog) with values of (p,1-p) equal to (.2,.8). These four models are presented in equation (57) of Chapter IV. A complete discussion of the models is given by Moser and Fei (1989b).

TABLE 2

RATIOS c_L / $M'(L_p)$ AND c_L / $M'(L_{1-p})$

M(x)	Logit	Probit	Skewed Logit	Loglog
p = .2	1.0	.94	.85	.92
1-p = .8	1.0	.94	1.12	.51

From Table 2, the logit version of SAM is optimal if the true expectation, M(x), is logit and nearly optimal when M(x) is probit. SAM's logit version is also nearly optimal at p = .2 when M(x) is loglog.

The first order asymptotic equivalence of the logit version of SAM and the two RM processes does not depend on the assumption $p_2 = 1 - p_1$. The results of this section hold for any $0 < p_1 < p_2 < 1$. A sketch of the proof is given below.

Let p_1, p_2, $0 < p_1 < p_2 < 1$, be the two constants used in SAM (9). Consider the following approximation to $G(x|\theta_1,\theta_2)$:

near L_{p_1} (j=1) (31)

$$(1 + \exp\{-\theta_2(x - \theta_1)\})^{-1} \cong p_1 + (x - L_{p_1})\cdot\lambda_{p_1} ,$$

near L_{p_2} (j=2) (32)

$$(1 + \exp\{-\theta_2(x - \theta_1)\})^{-1} \cong p_2 + (x - L_{p_2})\cdot\lambda_{p_2} ,$$

where $\lambda_{p_1} = \theta_2 p_1(1-p_1)$ and $\lambda_{p_2} = \theta_2 p_2(1-p_2)$. Note that for the two parameter logit model (16),

$$\lambda_{p_2} = \frac{p_2(1-p_2)}{p_1(1-p_1)}\cdot\lambda_{p_1} \quad \text{and} \tag{33}$$

$$L_{p_2} = L_{p_1} + (\lambda_{p_1})^{-1}\cdot p_1(1-p_1)\cdot \ln\left(\frac{p_2(1-p_1)}{p_1(1-p_2)}\right). \tag{34}$$

Substitute (33) and (34) into (32) to obtain

near L_{p_2} (35)

$$(1 + \exp(-\theta_2(x - \theta_1)))^{-1} \cong p_2 + (x - L_{p_1})\cdot c'\lambda_{p_1} - p_1(1-p_1)\cdot c'\cdot \ln\left(\frac{p_2(1-p_1)}{p_1(1-p_2)}\right),$$

where $c' = p_2(1-p_2) \;/\; p_1(1-p_1)$. Solving the likelihood equations (19) with the linear substitutions (31) and (35) yields

$$\begin{pmatrix} \hat{L}^{(n)}_{p_1} \\ \hat{L}^{(n)}_{p_2} \end{pmatrix} \sim \text{ASN}\left(\begin{pmatrix} L_{p_1} \\ L_{p_2} \end{pmatrix}, \begin{bmatrix} c_{L1}^{-2}\sigma_1^2/n(2c_{L1}^{-1}M'(L_{p_1})-1) & 0 \\ 0 & c_{L2}^{-2}\sigma_2^2/n(2c_{L2}^{-1}M'(L_{p_2})-1) \end{bmatrix}\right) \tag{36}$$

where $\sigma_i^2 = \lim_{X \longrightarrow L_{p_i}} \text{Var}(Y|x)$ for $i = 1,2$,

$c_{L1} = p_1(1-p_1)\{\ln[(1-p_1)/p_1] - \ln[(1-p_2)/p_2]\} \;/\; (L_{p_2} - L_{p_1})$

and $c_{L2} = p_2(1-p_2)\{\ln[(1-p_1)/p_1] - \ln[(1-p_2)p_2]\} \;/\; (L_{p_2} - L_{p_1})$. As before, these are optimal RM procedures if $M'(L_{p_1}) = c_{L1}$ and $M'(L_{p_2}) = c_{L2}$. If $M(x)$ is the two parameter logit model, then (36) simplifies to

$$\begin{pmatrix} \hat{L}^{(n)}_{p_1} \\ \hat{L}^{(n)}_{p_2} \end{pmatrix} \sim \text{ASN}\left(\begin{pmatrix} L_{p_1} \\ L_{p_2} \end{pmatrix}, \begin{bmatrix} \{n\theta_2^2 p_1(1-p_1)\}^{-1} & 0 \\ 0 & \{n\theta_2^2 p_2(1-p_2)\}^{-1} \end{bmatrix}\right). \tag{37}$$

Selecting p_1, p_2 Using the Two Parameter Logit Model (Symmetric Case)

In Chapter II, average minimum and minimax criteria for selecting $p_1,...,p_k$ in SAM were presented. These two approaches are now used to derive p_1, p_2 using the two parameter logit model (17).

Let $[a^*, 1 - a^*]$, $0 < a^* < 1/2$, define the range of interest for p^*. That is, the range of interest of the roots L_{p*} is the interval $[L_{a*}, L_{1-a*}]$. Since the logit model is symmetric over the range of interest, let $p_1 = p$ and $p_2 = 1 - p$. The problem reduces to selecting a single value p by the two different approaches.

Let p^* be a particular value in the range of interest, $[a^*, 1-a^*]$. From (9) and (17), the estimator $\hat{L}^{(n)}_{p*}$ is the solution to $p^* = (1 + \exp\{-\hat{\theta}^{(n)}_2(x - \hat{\theta}^{(n)}_1)\})^{-1}$. Since $G(x|\theta_1,\theta_2)$ from (17) is completely determined by L_p and L_{1-p}, the estimator, $\hat{L}^{(n)}_{p*}$, can be obtained by

$$\hat{L}^{(n)}_{p*} = r\cdot\hat{L}^{(n)}_{p} + (1-r)\cdot\hat{L}^{(n)}_{1-p} , \tag{38}$$

where $r = (1/2) + \ln\{(1-p^*)/p^*\} / 2\cdot\ln\{(1-p)/p\}$. From (30) and (38), when the true expectation is the two parameter logit model, the asymptotic variance of $\hat{L}^{(n)}_{p*}$ is

$$\sigma^{*2}_n(p^*,p) = \left(2n\theta_2^2 p(1-p)\right)^{-1}\left(1 + \frac{(\ln\{p^*/(1-p^*)\})^2}{(\ln\{p/(1-p)\})^2}\right). \tag{39}$$

The minimax solution is the value of p that minimizes the maximum $\sigma_n^{*2}(p^*,p)$ for $p^* \in [a^*,1-a^*]$. Note that for any given value of p,

$$\max_{p^* \in [a^*,1-a^*]} \sigma_n^{*2}(p^*,p) = \sigma_n^{*2}(a^*,p) \tag{40}$$

That is, the maximum value of $\sigma_n^{*2}(p^*,p)$ occurs at the boundaries of the interval $[a^*, 1 - a^*]$. Thus, the problem reduces to finding the p that minimizes

$$\big(p(1-p)\big)^{-1}\left(1 + \frac{(\ln\{a^*/(1-a^*)\})^2}{(\ln\{p/(1-p)\})^2}\right) \tag{41}$$

Column 2 of Table 3 gives optimal minimax values of p for various a^*.

Consider the average minimum approach of Chapter II. Let $\phi(\cdot)$ be the Uniform$[a^*, 1-a^*]$ density function. This is equivalent to assigning equal interest to each root, L_{p^*}, $p^* \in [a^*, 1-a^*]$. By (13) and (39), the average minimum solution is the value of p that minimizes

$$\int_{a^*}^{1-a^*} \big(2p(1-p)\big)^{-1}\left(1 + \frac{(\ln\{u/(1-u)\})^2}{(\ln\{p/(1-p)\})^2}\right) \partial u \; . \tag{42}$$

Column 3 of Table 3, labelled Avg-min, gives the average minimum values of p for various choices of a^*.

TABLE 3

MINIMAX AND AVERAGE MINIMUM P

a^*	Minimax p	Avg-min p
.05	.13	.19
.10	.15	.20
.15	.17	.22
.20	.19	.24
.25	.21	.26
.30	.23	.28
.40	.30	.34

Figure 5 on the next page graphically demonstrates the difference between the average minimum and minimax approaches. Each value of p produces a different $\sigma_n^{*2}(p^*,p)$ curve. In Figure 5, the function $\sigma_n^{*2}(p^*,p)$ is graphed for both p = .15 and p = .2. The curve $\sigma_n^{*2}(p^*,.15)$ has a smaller maximum for $p^* \in [.1,.9]$ than the curve $\sigma_n^{*2}(p^*,.2)$. However, the area under the $\sigma_n^{*2}(p^*,.2)$ curve is smaller. Thus, between these two possibilities, p = .15 is the minimax solution and p = .2 is the average minimum solution. As presented in Table 3, these are the minimax and average minimum solutions (when $a^* = .1$) over all values of p, $0 < p < 1/2$.

To obtain the results in Table 3, the two parameter logit model was used for $G(x|\theta)$. Thus, $\sigma_n^{*2}(p^*,p)$ was calculated using $G(x|\theta) = (1 + \exp\{-\theta_2(x-\theta_1)\})^{-1}$. If a different model is chosen for $G(x|\theta)$, then the minimax and average minimum values of p would change. Moser and Fei (1989b), consider the selection of p_1 and p_2 using four

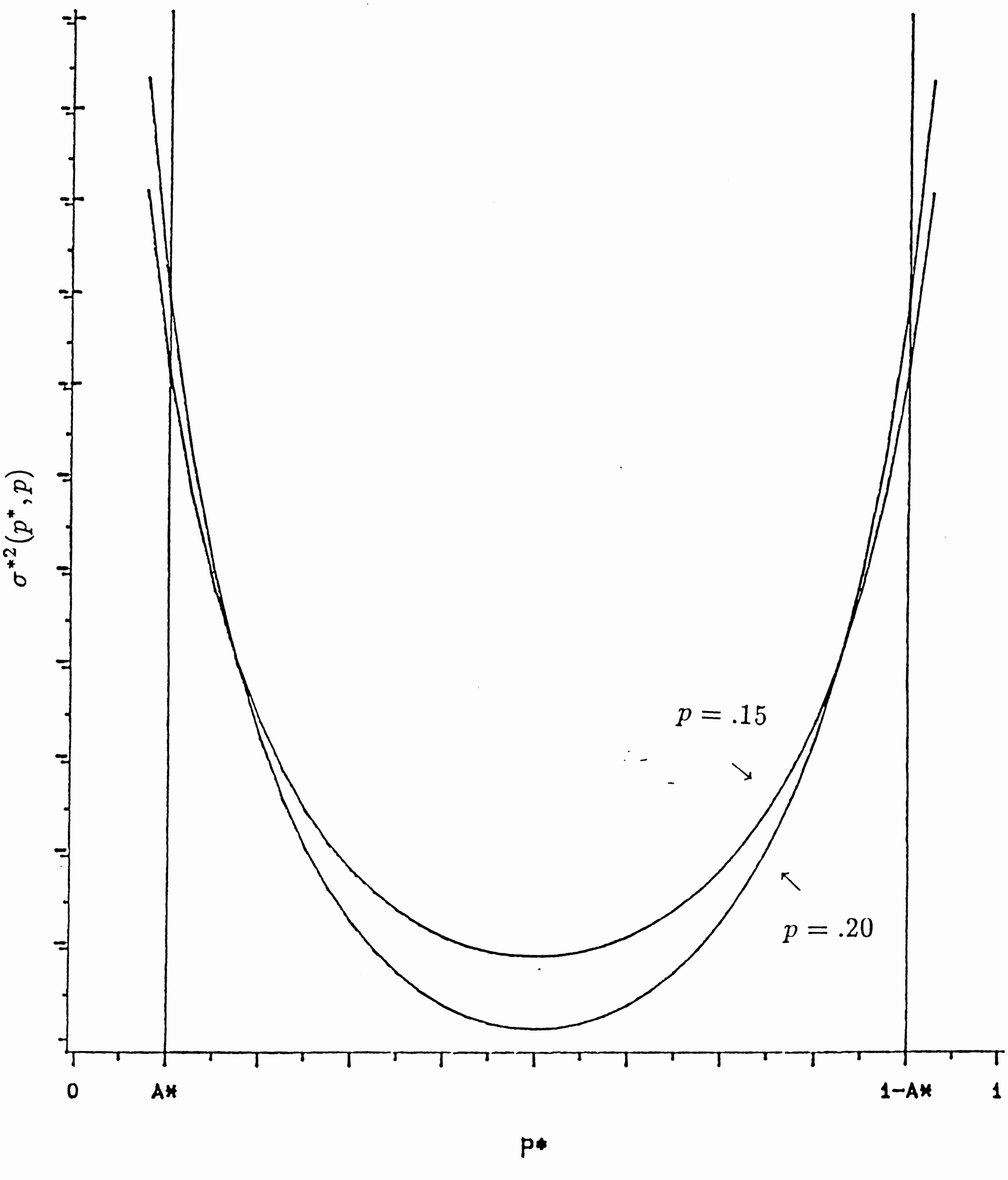

Figure 5. Minimax and Average Minimum p

different models (logit, probit, loglog, skewed logit) for $\mathcal{G}(x|\theta)$. Their discussion is based on a different method (a two dimensional Robbins Monro process). However, it is applicable to SAM because of the asymptotic equivalence of SAM and two independent RM processes.

The average minimum and minimax rules for $p_1,...,p_k$ are not only applicable to SAM, but are also valid for selecting $p_1,...,p_k$ for k independent RM procedures. For instance, Wetherill (1963) considered running two independent optimal RM procedures,

$$x_{n+1}^{(1)} = x_n^{(1)} - (n\cdot M'(L_p))^{-1}\cdot(y_n^{(1)} - p) \tag{43}$$
$$x_{n+1}^{(2)} = x_n^{(2)} - (n\cdot M'(L_{1-p}))^{-1}\cdot(y_n^{(2)} - (1-p)) \ ,$$

to obtain estimates of L_p and L_{1-p} for the logit model (17). Using these two estimates, an estimate of any root could be obtained from (38). He then demonstrated that the choice of $p = .2$ minimizes the product $\mathrm{Var}(\hat{\gamma}_1)\cdot\mathrm{Var}(\hat{\gamma}_2)$, where $\hat{\gamma}_1 = \hat{L}_{.5} = (.5)\cdot(x_n^{(1)} + x_n^{(2)})$ and $\hat{\gamma}_2 = [x_n^{(1)} + x_n^{(2)}] \ / \ [2\cdot\ln(p/(1-p)]$. From Table 3, a value of $p = .2$ is approximately the average minimum and minimax solution when the range of interest is (.1,.9) and (.2,.8), respectively.

Wetherill also demonstrated that the nonsequential design that minimizes the product of the asymptotic variances of the parameter estimates is to divide the design levels equally into two groups at $L_{.176}$ and $L_{.824}$. Since $L_{.176}$ and $L_{.824}$ are unknown before the experiment, using a sequential procedure to obtain design levels approaching these values is intuitively appealing.

Selecting p_1, p_2 Using the Two Parameter Logit Model (General Case)

In the previous section, the average minimum and minimax criteria were used to derive p_1 and p_2 using the logit model. The range of interest for p^* was $[a^*, 1-a^*]$, $0 < a^* < 1/2$. In many applications, however, the range of interest of p^* is not symmetric about .5. For instance, a researcher may be interested in estimating $L_{.95}$ and $L_{.99}$, the 95[th] and 99[th] percentiles of M(x), respectively. Let $[a_1, a_2]$, $0 < a_1 < a_2 < 1$, be the range of interest (where a_2 is not necessarily $1 - a_1$). In this section, the minimum avarage and minimax criteria will be used to select p_1 and p_2, where p_2 is not necessarily $1 - p_1$. The two parameter logit model is again used for $G(x|\theta)$.

As a generaliation of equation (38), the estimate of any L_p^* can be obtained from $\hat{L}_{p_1}$ and $\hat{L}_{p_2}$ by

$$\hat{L}_{p*} = r' \cdot \hat{L}_{p_1} + (1 - r') \cdot \hat{L}_{p_2} \,, \tag{44}$$

where

$$r' = \frac{\ln\{(1-p_2)/p_2\} - \ln\{(1-p^*)/p^*)}{\ln\{(1-p_2)/p_2\} - \ln\{(1-p_1)/p_1\}} \,.$$

The average minimum approach will be considered first. Assume that the roots from L_{a_1} to L_{a_2} are of equal interest. Therefore, let $\phi(\cdot)$ be the Uniform$[a_1, a_2]$ density function. Using (13), (37), and (44), assuming that M(x) is the two parameter logit model, the average minimum solution is the pair (p_1,p_2) that minimizes

$$\int_{a_1}^{a_2} \left[\frac{r'^2}{p_1\cdot(1-p_1)} + \frac{(1 - r')^2}{p_2\cdot(1-p_2)} \right] \partial p^* . \tag{45}$$

For a range of interest on p^* of [.95,.99] (that is $[a_1,a_2]$ = [.95,.99]), the expression (45) was calculated for a grid of (p_1,p_2) values (p_1 = .025 to .925 by .025, p_2 = p_1+ .05 to .975 by .025). The pair that produced the smallest value of (45) was (p_1,p_2) = (.15,.85). This results is somewhat surprising. Since the range of interest is in the upper tail, the minimum average p_1 and p_2 may be expected to be shifted toward the upper tail. Intuitively, using (p_1,p_2) = (.15,.95) is a more appropriate choice than (p_1,p_2) = (.05,.85).

To understand why (.15,.85) is the average minimum (p_1,p_2), recall that by (44), $\hat{L}_{p*}$ is a linear function of $\hat{L}_{p_1}$ and $\hat{L}_{p_2}$ $(r'\hat{L}_{p_1} + (1-r')\hat{L}_{p_2})$. Also, the asymptotic variance of $\hat{L}_p$ decreases as p approaches .5. When (p_1,p_2) = (.05,.85) is used to estimate $L_{p*} \in (L_{.95},L_{.99})$, the value of r′ from (44) satisfies (1-r′) > r′. Thus, more weight is placed on $\hat{L}_{.85}$, which has a smaller variance than $\hat{L}_{.05}$. When (p_1,p_2) = (.15,.95), more weight is placed on $\hat{L}_{.95}$, which has a larger asymptotic variance than $\hat{L}_{.15}$ (and $\hat{L}_{.85}$). Asymptotically, therefore, using (p_1,p_2) = (.05,.85) is superior to using (p_1,p_2) = (.15,.95). It is important to note that pairs (p_1,p_2), such as (.2,.8) and (.15,.95), produced only slightly larger values for (45) than the average minimum solution of (.05,.85).

The minimax solution in this general case is the pair (p_1,p_2) that minimizes the maximum $\sigma_n^{*2}(p^*,p_1,p_2)$ for $p^* \in [a_1,a_2]$. That is, the pair (p_1,p_2) that minimizes

$$\max_{p^* \in [a_1,a_2]} \left[\frac{r'^2}{p_1 \cdot (1-p_1)} + \frac{(1 - r')^2}{p_2 \cdot (1-p_2)} \right] \tag{46}$$

When $[a_1,a_2] = [.95,.99]$, the minimax pair, using the program described above, was found to be $(p_1,p_2) =$ (.075,.875). This is similar to the average minimum pair, $(p_1,p_2) =$ (.05,.85).

Estimating roots in the tails of a binary distribution with a small number of samples is a difficult task. As shown by Silvapulle (1981) and discussed in the first section of Chapter IV, MLEs do not exist when the responses are all 0's or all 1's, or when there is no overlapping in the responses. No overlapping in the responses occurs when the smallest design level with a response of 1 (0) is greater than the largest design level with a response of 0 (1). If all of the design levels fall in the upper tail of the distribution, then it is fairly likely that all of the responses will be 1's. Even if both 1's and 0's are observed, it is very possible that no overlap in the responses has occurred. If the design levels are in both tails, but not spread throughout the distribution, then it is again likely that no overlapping of the responses has occurred. In both of these situations, MLEs do not exist.

The average minimum pair (p_1,p_2) for estimating L_{p*}, $p^* \in [.95,.99]$ was found to be (.05,.85). Based on the discussion of the previous paragraph, (.05,.85) may not be a good choice of (p_1,p_2) for small or medium sample sizes. The simulation study below was designed to determine which pairs, (p_1,p_2), perform well in estimating L_{p*}, $p^* \in [.95,.99]$ for small and medium sized samples.

The performance of SAM was studied using the following four pairs of (p_1,p_2): (.2,.8), (.05,.85), (.15,.95), and (.4,.9). Each SAM procedure was given the same initial set of ten observations (as in Initial Procedure 1 of Chapter IV). A two parameter logit model ($\theta_1 = 0$, $\theta_2 = 1$) was used to generate the binary responses. The $\sqrt{}$MSEs of $\hat{L}^{(n)}_{.5}$, $\hat{L}^{(n)}_{.9}$, $\hat{L}^{(n)}_{.95}$, and $\hat{L}^{(n)}_{.99}$ are reported in Table 4 on the following page.

The average minimum solution for a range of interest on p^* of [.95,.99] was found to be (p_1,p_2) = (.05,.85). In this simulation study, (p_1,p_2) = (.05,.85) generated the smallest MSEs for estimating $L_{.99}$. For estimating $L_{.95}$, the pairs (.05,.85) and (.2,.8) produced the lowest MSEs. However, for estimating $L_{.5}$ and $L_{.9}$, (.05,.85) did not perform as well as using (p_1,p_2) = (.2,.8). Using (.2,.8) compared reasonably well to the other choices in every situation.

TABLE 4

MONTE CARLO $\sqrt{MSE}$ FOR ESTIMATING L_{p*} USING SAM

		(p_1,p_2)			
p^*	/ n	(.2,.8)	(.05,.85)	(.15,.95)	(.4,.9)
.5	n = 10	.5339	.6099	.5705	.5395
	15	.4645	.5836	.5651	.4807
	20	.3940	.5107	.4254	.3871
	30	.3178	.4263	.3789	.3177
.9	n = 10	1.1174	1.2510	1.2344	1.1861
	15	.9897	1.0214	1.0332	.9810
	20	.8171	.8488	.8465	.8173
	30	.6692	.7071	.7115	.6542
.95	n = 10	1.3626	1.4954	1.4821	1.4656
	15	1.2554	1.2196	1.2644	1.2393
	20	1.0253	1.0439	1.0616	1.0546
	30	.8446	.8624	.8835	.8456
.99	n = 10	1.7267	1.7102	1.7615	1.7835
	15	1.6914	1.5413	1.6244	1.6698
	20	1.4358	1.3728	1.4060	1.4836
	30	1.2360	1.1782	1.1955	1.2649

Asymptotic Variance and Bias

In this section, the asymptotic variances of estimators from the logit version of SAM are presented. To study the robustness of SAM, the asymptotic variances and biases are also derived when the true expectation of Y, M(x), is not the two parameter logit model. The derivation depends upon the first order equivalence of the logit version of SAM and a two dimensional RM procedure. Thus, the discussion begins by introducing the two dimensional RM process. Consider estimating L_{p*} using two independent RM procedures (with n observations each), one to estimate L_p and the other to estimate L_{1-p}, both using the optimal $n \cdot A_n = M'(L_p)^{-1} = M'(L_{1-p})^{-1}$. Using the two parameter logit model, an estimate of any root L_{p*} can be constructed as in (38).

First, consider the case where the expectation of Y, M(x), is the two parameter logit model. By (5) and (38), the asymptotic variance of $\hat{L}_{p*}$ obtained from the two independent RM procedures is

$$\{2n\theta_2^2 p(1-p)\}^{-1}\left[1 + \frac{\ln\{p^*/(1-p^*)\}^2}{\ln\{p/(1-p)\}^2}\right]. \tag{47}$$

Earlier in this chapter, a first order approximation to SAM was shown to be asymptotically equivalent to two independent RM procedures. Therefore, the expression in (47) is also the asymptotic variance of $\hat{L}_{p*}$ from the logit version of SAM. For the sample sizes and values of p^* used in the simulation study of Chapter IV, (47) is evaluated and presented in the last row of Tables 8 and 9.

When the true expectation of Y, M(x), is not the two parameter logit model, then estimators from either procedure may be biased. Recall that the consistency of X_n from the RM procedure (4) is not dependent upon the true expectation of Y. Thus, the consistency of $\hat{L}_p$ and $\hat{L}_{1-p}$ from both the two independent RM procedures and SAM (by the first order asymptotic equivalence) holds regardless of M(x). However, the estimate of any root L_{p*} for $p^* \neq p, 1-p$, using (38), is based on the two parameter logit model. If M(x) is not the two parameter logit model, then the constant, r', used in

(38) will be incorrect, and a biased estimate will be produced. Thus, even though $\hat{L}_p$ and $\hat{L}_{1-p}$ are consistent estimators of L_p and L_{1-p}, the estimator of any other L_{p*} may not be consistent.

To generalize the above discussion, consider running two independent RM procedures to estimate L_{p*}, as before. However, use $p = p_1$ for one procedure and $p = p_2$, $p_2 > p_1$ for the other, with p_2 not necessarily $1-p_1$. Estimate L_{p*} with

$$\hat{L}_{p*} = k_a \cdot \hat{L}_{p_1} + (1-k_a) \cdot \hat{L}_{p_2}, \tag{48}$$

where k_a is a constant determined by a selected model, $G^*(x|\theta)$. For instance, if $G^*(x|\theta)$ is the two parameter logit model, then from (44)

$$k_a = \frac{\ln\{(1-p_2)/p_2\} - \ln\{(1-p^*)/p^*)}{\ln\{(1-p_2)/p_2\} - \ln\{(1-p_1)/p_1\}}. \tag{49}$$

Let k_t be the corresponding constant for the true expectation of Y. Since $\hat{L}_{p_1} \text{----->} L_{p_1}$ and $\hat{L}_{p_2} \text{----->} L_{p_2}$, $\hat{L}_{p*}$ converges to $k_a \cdot L_{p_1} + (1-k_a) \cdot L_{p_2}$. The bias of L_{p*} is then $(k_t - k_a)(L_{p_2} - L_{p_1})$. Consider using the two independent RM procedures with $p_1 = .2$, $p_2 = .8$ and the two parameter logit model for $G^*(x|\theta)$. Table 5 presents the biases of $\hat{L}_{p*}$, $p^* = .25,.5,.75$, when M(x), the true expectation of Y, follows one of three different models. The probit, skewed logit and loglog models used in Table 5 are described in equation (57) of Chapter IV.

TABLE 5

BIAS OF $\hat{L}_{p*}$

Model	p^* .25	.5	.75
Probit (θ_1= -.25,θ_2= 2)	.0202	0	-.0202
Skewed Logit (θ_1= -1,θ_2= .7)	.0289	.1158	.0376
Loglog (θ_1= 0,θ_2= .5)	-.0976	-.2912	-.1118

By (36), these biases are also appropriate for $\hat{L}_{p*}$ from the logit version of SAM.

Fei (1989) and Moser and Fei (1989b) both present detailed discussions of the asymptotic variance and biases of L_{p*} from using two independent RM procedures. Formulas and calculations are presented for the logit, probit, skewed logit and loglog models. The following development of the mean square error (MSE) of $\hat{L}_{p*}$ follows their approach.

The optimal values for $n \cdot A_n$ for the two independent RM procedures are $M'(L_{p_1})^{-1}$ and $M'(L_{p_2})^{-1}$. Let $A_1^{(a)}$ and $A_2^{(a)}$ be the corresponding values using $G^*(x|\theta)$. If $A_1^{(a)} \neq M'(L_{p_1})^{-1}$ or $A_2^{(a)} \neq M'(L_{p_2})^{-1}$, then the RM procedures will not be optimal. Using (5) and (48), the asymptotic variance of $\hat{L}_{p*}$ using the two independent RM procedures is

$$(1/n) \cdot \Big(k_a^2 p_1 (1-p_1) A_1^{(a)^2} (2A_1^{(a)} \cdot M'(L_{p_1}) - 1)^{-1} + \tag{50}$$

$$(1-k_a)^2 p_2 (1-p_2) A_2^{(a)^2} (2A_2^{(a)} \cdot M'(L_{p_2}) - 1)^{-1} \Big) .$$

If $A_1^{(a)} = M'(L_{p_1})^{-1}$, $A_2^{(a)} = M'(L_{p_2})^{-1}$, $p_1 = p$, $p_2 = 1 - p$, and $M(x)$ is the two parameter logit model, then equation (50) simplifies to the expression presented in (47).

Let $L_p^{(a)}$ be the p^{th} percentile of the selected model, $G^*(x|\theta)$. Recall that $\hat{L}_{p_1}$, $\hat{L}_{p_2}$ from the two RM procedures converge to L_{p_1}, L_{p_2} respectively, regardless of the selected model $G^*(x|\theta)$. Since the convergence of $\hat{L}_{p_1}$, $\hat{L}_{p_2}$ is independent of the true expectation of Y, $\hat{L}_{p_1}$, $\hat{L}_{p_2}$ also converge to $L_{p_1}^{(a)}$, $L_{p_2}^{(a)}$. Thus, in this notation, $(L_{p_1}^{(a)}, L_{p_2}^{(a)}) = (L_{p_1}, L_{p_2})$. Consider the MSE of $\hat{L}_{p*}$,

$$MSE(\hat{L}_{p*}) = E\{\hat{L}_{p*} - L_{p*}\}^2 = E\{\hat{L}_{p*} - L_{p*}^{(a)}\}^2 + (L_{p*}^{(a)} - L_{p*})^2 + 2\cdot\{L_{p*}^{(a)} - L_{p*}\}\cdot E\{\hat{L}_{p*} - L_{p*}^{(a)}\}. \tag{51}$$

Now, $\hat{L}_{p*}$ converges to $k_a \cdot L_{p_1} + (1-k_a)\cdot L_{p_2} = L_{p*}^{(a)}$. Therefore, the MSE of $\hat{L}_{p*}$ converges to

$$MSE(\hat{L}_{p*}) = Var(\hat{L}_{p*}) + (L_{p*}^{(a)} - L_{p*})^2 = Var(k_a\hat{L}_{p_1} + (1-k_a)\hat{L}_{p_2}) + (k_a - k_t)^2(L_{p_1} - L_{p_2})^2. \tag{52}$$

Note that the MSE of $\hat{L}_{p*}$ includes both a variance term and a bias term. Note that the bias term of (52) is the square of the biases listed in Table 5. Using $p_1 = .2$ and $p_2 = .8$, Table 6 presents the MSEs of $\hat{L}_{p*}$ under the four models used for M(x) in the simulation study of Chapter IV. In each case, the two parameter logit model was used to construct the estimators.

TABLE 6

ROBBINS-MONRO, SAM ASYMPTOTIC MSEs

True Model:

MSE:	Logit	Loglog	Skewed Logit	Probit
$\hat{L}_{.25}$	$\frac{5.085}{n}$	$\frac{17.222}{n} + .0095$	$\frac{5.609}{n} + .0007$	$\frac{6.664}{n} + .0004$
$\hat{L}_{.5}$	$\frac{3.125}{n}$	$\frac{7.039}{n} + .0851$	$\frac{4.606}{n} + .0134$	$\frac{4.095}{n}$
$\hat{L}_{.75}$	$\frac{5.088}{n}$	$\frac{5.686}{n} + .0133$	$\frac{9.381}{n} + .0014$	$\frac{6.664}{n} + .0004$

These values are calculated and presented in the last row of the appropriate simulation tables of Chapter IV (Tables 8 - 15). Consult Moser and Fei (1989b) for a more detailed discussion of the MSE values.

CHAPTER IV

SIMULATION STUDY

The Setup

In the first three chapters, asymptotic properties of the various procedures have been presented. Under certain conditions, Robbins-Monro, Anbar, SAM, and Wu's procedures are asymptotically equivalent. In this chapter, a simulation study is performed to compare the procedures for small and medium binary data samples. The procedures will be evaluated on their ability to estimate a single root, L_p, and multiple roots.

Recall that Robbins-Monro's (4), Anbar's (7), and Wu's (8) procedures were designed to estimate a single root, L_p, of $M(x) = p$. Two methods of extending these procedures to estimate any number of roots are considered. First, two independent versions of the same procedure (either two independent Robbins-Monro, two independent Anbar, or two independent Wu procedures) can be run to obtain estimates of L_{p_1} and L_{p_2}. Then, using the two parameter logit model, an estimate of any L_{p*} can be obtained from (44). Based on the results in Chapter III, $p_1 = .2$ and $p_2 = .8$ are used in this simulation study.

As a second extension, Anbar's and Wu's procedures can be run as designed (estimating $L_{.5}$) and the slope estimator in combination with $\hat{L}_{.5}$ can be used to estimate any L_{p^*}. For Anbar's procedure, again using the two parameter logit model,

$$\hat{L}^{(n)}_{p*} = \hat{L}^{(n)}_{.5} + (1 / 4 \cdot b_n) \cdot \ln\{p^*/(1-p^*)\} \ , \tag{53}$$

where b_n is given in (6) with $m(n) = 1$. For Wu's procedure,

$$\hat{L}^{(n)}_{p*} = \hat{L}^{(n)}_{.5} + (\hat{\theta}^{(n)}_2)^{-1} \cdot \ln\{p^*/(1-p^*)\}. \tag{54}$$

The RM procedure is not extended by the second method since a slope estimator is never calculated from the data during the process.

As mentioned in Chapter II, bounded versions of sequential approximation procedures have been found to improve their performance for small to medium sample sizes. The following truncated versions of the procedures, in which the step size from X_n to X_{n+1} is bounded, are used in this study. For Anbar's procedure the updating rule is

$$X_{n+1} = X_n - (h_n/n)(Y_n - p) \ , \tag{55}$$

where $h_n = \max\{\delta_1, \min(\delta_2, b_n^{-1})\}$, $0 \le \delta_1 < \delta_2$. Wu defined the following truncated version of (8). Let d_n be the solution of $X_{n+1} = X_n - (d_n/n) \cdot (Y_n - p)$, where $X_{n+1} = \hat{\theta}^{(n)}_1 - (1 / \hat{\theta}^{(n)}_2) \cdot \ln\{(1-p)/p\}$. The $(n+1)^{st}$ design point is chosen by the rule

$$X_{n+1} = X_n - (d_n^* / n)\cdot(Y_n - p) \ , \tag{56}$$

where $d_n^* = \max[\delta_1, \min(d_n, \delta_2)]$, for $0 \le \delta_1 < \delta_2$. The truncated version of SAM is given in Chapter II by (15).

The following six basic procedures, all with a total of $2\cdot n$ design points, are considered:

1) SAM-δ_2 -- procedure (15) with upper bound δ_2, where L_{p*} is estimated by (38);

2) RM-w -- two independent versions of procedure (4), with n updates each, one with p=.2 and the other with p=.8, both with $A_n = w/n$, where L_{p*} is estimated by (38);

3) AN1-δ_2 -- procedure (55) with p = .5, where L_{p*} is estimated by (53);

4) AN2-δ_2 -- two independent versions of procedure (55) using p = .2 and p = .8, with n updates each, where L_{p*} is estimated by (38);

5) WU1-δ_2 -- procedure (56) with p = .5, where L_{p*} is estimated by (54);

6) WU2(δ_2) -- two independent versions of procedure (56) using p = .2 and p = .8, with n updates each, where L_{p*} is estimated by (38).

Since SAM, WU1, and WU2 require the existence of MLEs at each update, a starting procedure is needed to generate the design levels until MLEs exist. Two different initial procedures are considered. The first has five fixed initial design levels (a total of 10 intitial observations), and samples for which MLEs do not exist are discarded. The second uses a Robbins-Monro procedure to generate initial design levels until MLEs exist.

Silvapulle (1981) provided existence conditions for MLEs from binary distributions. Let $x^{+}_{max(min)}$ = $max(min)\{x_{ij}:y_{ij}= 1\}$ and $x^{-}_{max(min)} = max(min)\{x_{ij}:y_{ij}= 0\}$. For the two parameter logit model, if

1) $(x^{+}_{min},x^{+}_{max}) \cap (x^{-}_{min},x^{-}_{max}) \neq \emptyset$,

2) $x^{+}_{min} < x^{-}_{min} = x^{-}_{max} < x^{+}_{max}$, or

3) $x^{-}_{min} < x^{+}_{min} = x^{+}_{max} < x^{-}_{max}$,

then the MLEs of θ_1 and θ_2 exist and are unique. Thus, for existence and uniqueness of MLEs, there must be some overlapping of the responses. If MLEs exist at the n^{th} update, then they exist at each subsequent update. Therefore, once the existence conditions have been satisfied, design levels can be calculated by SAM's and Wu's updating rules for every future update.

In order to evaluate the robustness of the procedures, four different parametric models for the expectation M(x) are used to generate the binary responses.

$$\text{logit:} \quad 1 / (1 + \exp(-\theta_2(x-\theta_1)) \quad \text{with } \theta_1 = 0, \ \theta_2 = 1 \tag{57}$$

$$\text{probit:} \quad \Phi\{(x-\theta_1)/ \theta_2\} \quad \text{with } \theta_1 = -.25, \ \theta_2 = 2$$

log-log: $1 - \exp[-\exp\{\theta_2(x-\theta_1)\}]$

with $\theta_1 = 0$, $\theta_2 = .5$

skewed logit: $[1 + \exp\{-\theta_2(x-\theta_1)\}]^{-2}$

with $\theta_1 = -1$, $\theta_2 = .7$

Regardless of the true model, the MLEs in SAM, WU1, and WU2 were calculated using the two parameter logit model. In the simulation program, all procedures use the same set of random numbers for each set of trials.

Recall that the optimal A_n for the RM procedure (4) is $A_n = (n \cdot M'(L_p))^{-1}$. For the models in this simulation study (57), Table 7 presents the optimal values of $n \cdot A_n$ when p = .2, .5, and .8.

TABLE 7

OPTIMAL $n \cdot A_n$

p	Model: Logit	Probit	Loglog	Skewed Logit
.2	6.25	7.14	11.24	6.45
.5	4.00	5.03	5.78	4.88
.8	6.25	7.14	6.25	8.46

Initial procedure 1

In the first initial design, ten binary observations are generated at five fixed design levels. If the MLEs exist, the starting values for each procedure are calculated using $\hat{\theta}_1^{(10)}$, $\hat{\theta}_2^{(10)}$. For example, the design levels at the 11^{th} update for SAM, RM, AN2, and WU2 are $x_{11,1} = \hat{\theta}_1^{(10)} - (\hat{\theta}_2^{(10)})^{-1} \cdot \ln\{(1-.2)/.2\}$ and $x_{11,2} = \hat{\theta}_1^{(10)} + (\hat{\theta}_2^{(10)})^{-1} \cdot \ln\{(1-.2)/.2\}$. For AN1 and WU1, the 11^{th} design level is $x_{11} = \hat{\theta}_1^{(10)}$. Once the starting levels have been obtained, the individual procedures are used to calculate design levels for the remaining updates. If the MLEs do not exist for the initial sample of ten observations, or if $\hat{\theta}_2^{(10)} \le 0$, then the sample is discarded. Two sets of ten initial design levels are used:

Set 1: Levels $L_{.1}, L_{.3}, L_{.5}, L_{.7}, L_{.9}$ with 1,2,4,2, and 1 observations, respectively ;

Set 2: Levels $L_{.3}, L_{.46}, L_{.56}, L_{.66}, L_{.8}$ with 1,2,4,2, and 1 observations, respectively.

Upper bounds of δ_2 = 10, 50, and 100, and a lower bound of δ_1 = 0 were selected. The larger values of δ_2 allow larger steps from X_n to X_{n+1} at each update. A total of 500 samples are generated, including the discarded samples. The MSE is calculated by $\left(\sum(\hat{L}_{p*} - L_{p*})^2 / n'\right)^{1/2}$, where $\hat{L}_{p*}$ is the estimate from the given procedure, L_{p*} is the true $(p^*)^{th}$ percentile from the appropriate model (57), and the summation is over the n′ nondiscarded samples.

Results of Initial Procedure 1

Tables 8 - 15 present the MSEs for the four different models using each set of design levels. Each table provides the MSEs of $\hat{L}_{.5}$ and $\hat{L}_{.75}$.

The MSEs from SAM for estimating $L_{.75}$ are less than those from any other procedure in almost every situation. In a few situations, WU1 or WU2 produced MSEs of similar size as SAM. For estimating $L_{.5}$, SAM's only competition comes from WU1. In general, WU1 has smaller MSEs when n = 15 and n = 20. This advantage does not hold, for larger (n = 30) sample sizes, where the MSEs for WU1 and SAM are similar in size. These patterns hold for all four of the models used to generate the binary responses.

SAM, WU1, and WU2 use a logit model to calculate the new design levels at each update. However, they generally continued to outperform AN1, AN2, and RM, even when a skewed model (such as the loglog or skewed logit) generated the binary responses. One exception to this rule occured for the loglog model with initial set 2. For n = 10 and n = 15, AN2 (δ_2= 10) and RM-6 produced MSEs of similar size as WU1 and SAM for estimating $L_{.5}$.

For estimating $L_{.5}$ with the RM procedures, RM-6 is superior to RM-1 and RM-36. For estimating $L_{.75}$, both RM-6 and RM-36 produce lower MSEs than RM-1. Several times, the MSEs from the RM-1 procedure increased when more observations were taken. From Table 7, RM-6 is closer to the optimal RM procedure than RM-1 or RM-36 in each situation.

Note that SAM's performance is not affected greatly by the choice of bounds. WU1 and WU2 also remain fairly stable over the various bounds. On occasion, WU1 and WU2 performed poorer when the bounds were too restrictive ($\delta_2 = 10$). For AN1 and AN2, the the tighter bounds were often superior.

Recall that AN1 and WU1 are designed to estimate a single root ($L_{.5}$ in this study). By choosing the design levels around $L_{.2}$ and $L_{.8}$ instead of a single value, AN2 and WU2 were constructed to provide better estimates of the roots throughout the entire curve, M(x). As expected, WU1 outperformed its counterpart (WU2) in estimating $L_{.5}$. However, WU2 did not always prove to be better in estimating $L_{.75}$. For small sample sizes, WU1 even produced better $L_{.75}$ estimates than WU2. AN2 proved to be superior to AN1 for estimating both $L_{.5}$ and $L_{.75}$.

In order to compare the simulation and asymptotic results, the asymptotic variance of $\hat{L}^{(n)}_{p*}$ from the RM procedure is presented in the last row of each table. The asymptotic values were derived in the last section of Chapter III. Due to the first order asymptotic equivalence of SAM and the RM process, these values will also be the asymptotic variances of $\hat{L}^{(n)}_{p*}$ from SAM.

For the logit model, the MSEs from SAM's $L_{.5}$ estimator were very similar to their asymptotic value for all sample sizes. For $L_{.75}$, the MSEs were similar for n = 20 and n =

30. Note that the MSEs from SAM proved to be fairly similar to the asymptotic value for the other three models as well. The asymptotic variances of $\hat{L}_{.5}$ from WU1 and AN1 are less than for SAM and RM. However, the simulation results show that they approach their asymptotic values much more slowly than SAM.

The number of discarded samples was not small. It ranged from 136 to 157 of the 500 samples. By selecting the initial design levels in a different manner, the proportion of discarded samples could be reduced. The results of this section are based on those samples for which MLEs existed after ten initial observations.

Initial Procedure 2

In the second initial procedure, a RM procedure is used to generate the initial design levels. Three values of $n \cdot A_n$ (1, 6, and 36) and three starting design levels (design levels for the first update) are used. From Table 7, the $n \cdot A_n$ values of 1 and 36 represent over and underestimates of the asymptotically optimal values, respectively. The initial step sizes using $n \cdot A_n = 36$ will be much larger than using $n \cdot A_n = 1$.

For SAM, AN2, and WU2, two independent RM procedures, one with $p = .2$ and the other with $p = .8$, are used to calculate the initial design levels. Thus, two starting

TABLE 8

MONTE CARLO √MSEs FOR ESTIMATING L_{p^*}

MODEL: LOGIT $\theta_1 = 0$, $\theta_2 = 1$

INITIAL SET 1

(143 discarded samples)

p^*	.5				.75			
N	10	15	20	30	10	15	20	30
RM-1	.7208	.7819	.7318	.7221	1.1939	1.2400	1.2819	1.2135
RM-6	.6168	.5996	.5098	.4560	1.0393	.9813	.9629	.7847
RM-36	.9419	.8078	.6935	.6245	1.2308	1.0378	.9297	.7787
AN1-10	.7878	.6065	.5367	.4240	1.3565	1.0993	.9967	.7977
AN1-50	.7508	.6232	.5342	.4383	1.0625	1.1168	1.0025	.7217
AN1-100	.7664	.6316	.5436	.4458	1.0649	1.1474	1.0061	.7428
AN2-10	.6153	.5874	.4900	.4444	1.0123	.9124	.8282	.6929
AN2-50	.6120	.5765	.4875	.4359	1.0625	.9370	.8299	.7133
AN2-100	.6129	.5774	.4872	.4399	1.0649	.9415	.8328	.7266
WU1-10	.6124	.4052	.3618	.3071	.8972	.6885	.6775	.6063
WU1-50	.5062	.4184	.3654	.2972	.8182	.7141	.6738	.5969
WU1-100	.5095	.4180	.3648	.2993	.8385	.7075	.6733	.6094
WU2-10	.6389	.5643	.4690	.3859	1.0486	.8799	.7639	.5698
WU2-50	.6259	.5173	.4490	.3626	.8870	.6512	.5471	.4719
WU2-100	.6227	.5159	.4577	.3735	.8785	.6399	.5644	.4940
SAM-10	.5344	.4749	.4182	.3329	.7351	.6505	.5524	.4429
SAM-50	.5263	.4621	.4047	.3189	.7271	.6445	.5370	.4403
SAM-100	.5169	.5159	.3901	.3158	.7292	.6361	.5355	.4393
Asympt. RM	.5590	.4564	.3953	.3227	.7133	.5824	.5044	.4118

TABLE 9

MONTE CARLO √MSEs FOR ESTIMATING L_{p^*}

MODEL: LOGIT $\theta_1 = 0$, $\theta_2 = 1$

INITIAL SET 2

(150 discarded samples)

p^*	.5				.75			
N	10	15	20	30	10	15	20	30
RM-1	.5493	.6101	.5925	.5319	1.1449	1.1865	1.1609	1.1871
RM-6	.5730	.6454	.6474	.5972	1.1803	1.2414	1.2431	1.2897
RM-36	.5777	.6521	.6580	.6098	1.1868	1.2512	1.2581	1.3081
AN1-10	1.0454	.7121	.7026	.5771	1.3625	1.1251	1.0900	.9636
AN1-50	1.0722	.7564	.7181	.5934	1.3949	1.1979	1.0872	.9459
AN1-100	1.1581	.8254	.7569	.6102	1.4729	1.2677	1.1257	.9856
AN2-10	.5017	.4953	.4467	.3926	.9938	.9059	.7889	.7299
AN2-50	.6242	.5801	.5442	.4804	1.1026	.9659	.8657	.7650
AN2-100	.6482	.6113	.5792	.4976	1.1225	1.0006	.8944	.7823
WU1-10	.5762	.3384	.2978	.3629	.8001	.6635	.6662	.7299
WU1-50	.3795	.3356	.2963	.3622	.6522	.6617	.6669	.7226
WU1-100	.3748	.3387	.2933	.3614	.6513	.6710	.6641	.7176
WU2-10	.5640	.5416	.4819	.4521	1.0134	.8974	.7545	.6629
WU2-50	.7562	.5787	.4872	.4305	.9328	.7172	.5855	.5225
WU2-100	.7682	.5857	.5092	.4184	.9587	.7439	.6091	.5071
SAM-10	.5372	.4417	.3759	.3181	.7870	.6744	.5475	.4477
SAM-50	.4800	.4349	.3867	.3038	.7259	.6149	.5406	.4405
SAM-100	.4828	.4512	.3861	.3033	.7255	.6181	.5421	.4479
Asympt. RM	.5590	.4564	.3953	.3227	.7133	.5824	.5044	.4118

TABLE 10

MONTE CARLO $\sqrt{MSE}$s FOR ESTIMATING L_{p*}

MODEL: PROBIT $\theta_1 = -.25$ $\theta_2 = 2$

INITIAL SET 1

(139 discarded samples)

p^*	.5				.75			
N	10	15	20	30	10	15	20	30
RM-1	.8111	.8668	.8303	.7979	1.3636	1.4078	1.4344	1.3624
RM-6	.8442	.9240	.9044	.8893	1.4060	1.4776	1.5310	1.4827
RM-36	.8499	.9342	.9179	.9063	1.4135	1.4899	1.5482	1.5046
AN1-10	.9651	.7779	.6465	.5251	1.5058	1.2937	1.1453	.9782
AN1-50	.9176	.8020	.6447	.5391	1.8273	1.4725	1.1821	.9066
AN1-100	.9569	.8187	.6655	.5406	1.8224	1.5167	1.1971	.9324
AN2-10	.6812	.6298	.5513	.4693	1.1290	.9939	.8919	.7246
AN2-50	.7148	.6227	.5487	.4643	1.2296	1.0033	.9165	.7374
AN2-100	.7225	.6315	.5563	.4694	1.2578	1.0250	.9396	.7592
WU1-10	.7194	.4931	.4350	.3762	1.5836	1.2919	1.2858	1.1997
WU1-50	.6116	.5079	.4454	.3692	.9509	.8777	.8096	.7248
WU1-100	.6169	.5100	.4454	.3693	.9740	.8714	.8169	.7290
WU2-10	.7017	.6136	.5247	.4392	1.1819	.9877	.8445	.6223
WU2-50	.7456	.6135	.5255	.4278	.9968	.7359	.6390	.5641
WU2-100	.7633	.5997	.5221	.4353	1.0355	.7225	.6335	.5800
SAM-10	.6171	.5513	.4812	.3795	.9135	.7502	.6290	.5064
SAM-50	.6238	.5321	.4561	.3621	.9095	.7574	.6237	.5052
SAM-100	.6202	.5327	.4556	.3636	.9174	.7571	.6194	.5078
Asympt. RM	.6399	.5225	.4525	.3695	.8163	.6665	.5772	.4713

TABLE 11

MONTE CARLO √MSEs FOR ESTIMATING L_{p*}

MODEL: PROBIT $\theta_1 = -.25$ $\theta_2 = 2$

INITIAL SET 2

(136 discarded samples)

p^*	.5				.75			
N	10	15	20	30	10	15	20	30
RM-1	.9779	.9812	.9100	.8299	1.5199	1.5745	1.4460	1.3994
RM-6	.8483	.7726	.6223	.5042	1.3196	1.2216	1.0076	.8547
RM-36	1.0555	.9070	.7819	.6506	1.3166	1.1022	1.0047	.8165
AN1-10	1.1936	.9158	.8657	.6756	1.6098	1.4068	1.3526	1.1863
AN1-50	1.2072	.9568	.8805	.6946	1.7557	1.5716	1.3653	1.1733
AN1-100	1.2724	.9996	.9109	.7156	1.8099	1.5834	1.3973	1.2411
AN2-10	.8003	.6950	.5605	.4613	1.2431	1.0689	.8657	.7037
AN2-50	.8277	.6870	.6161	.5444	1.3289	1.0810	.9448	.7781
AN2-100	1.0418	.8207	.7247	.6616	1.6301	1.2570	1.1371	.9128
WU1-10	.6612	.4409	.3818	.4216	.9212	.8084	.7977	.7921
WU1-50	.5010	.4354	.3744	.4856	.7775	.7891	.7838	.8749
WU1-100	.5058	.4427	.3809	.6118	.7798	.8160	.7896	.9774
WU2-10	.8958	.8624	.7048	.6387	1.3887	1.3521	1.0888	.9618
WU2-50	1.0220	.9011	.7320	.6700	1.3769	1.2661	.9852	.9134
WU2-100	1.0934	.9027	.7539	.6694	1.4526	1.2812	.9990	.9168
SAM-10	.5566	.5044	.4455	.3681	.8410	.7697	.6500	.5174
SAM-50	.5510	.4948	.4489	.3550	.7907	.8433	.6530	.5208
SAM-100	.5664	.4980	.4379	.3624	.7807	.7505	.6520	.5305
Asympt. RM	.6399	.5225	.4525	.3695	.8163	.6665	.5772	.4713

TABLE 12

MONTE CARLO √MSEs FOR ESTIMATING L_{p^*}

MODEL: SKEWED LOGIT $\theta_1 = -1$ $\theta_2 = .7$

INITIAL SET 1

(143 discarded samples)

p^*	.5				.75			
N	10	15	20	30	10	15	20	30
RM-1	.8358	.8821	.8435	.8269	1.2323	1.2595	1.2869	1.2226
RM-6	.7191	.6757	.6004	.5265	1.0577	.9792	.9314	.8042
RM-36	1.0957	.9116	.8006	.7100	1.4913	1.2428	1.1017	.9585
AN1-10	.9061	.7971	.6466	.5171	1.3772	1.2685	1.1203	.9350
AN1-50	.9391	.8077	.6534	.5274	1.7308	1.4107	1.1928	.9037
AN1-100	1.0090	.8182	.6580	.5293	1.8041	1.4443	1.2318	.9031
AN2-10	.7055	.6541	.5796	.5065	1.0363	.9212	.8528	.7417
AN2-50	.7288	.6666	.5933	.5207	1.1726	1.0295	.9659	.8352
AN2-100	.7379	.6716	.5968	.5351	1.2135	1.0373	.9688	.8670
WU1-10	.6578	.4998	.4264	.3647	1.0059	.8942	.8396	.7542
WU1-50	.6110	.5129	.4524	.3629	.9473	.8984	.8620	.7431
WU1-100	.6083	.5167	.4790	.3572	.9576	.8964	.8605	.7365
WU2-10	.7382	.6538	.5936	.4731	1.0858	.9335	.8572	.6648
WU2-50	.8198	.6498	.5816	.4715	1.1366	.8451	.7903	.6700
WU2-100	.8338	.6347	.5875	.4755	1.1924	.8341	.7975	.6805
SAM-10	.6321	.5828	.4958	.4147	.8772	.8673	.7189	.5863
SAM-50	.6483	.5799	.4882	.4054	.8629	.8639	.6975	.5978
SAM-100	.6385	.5811	.4852	.4135	.8695	.8543	.7022	.6045
Asympt. RM	.6787	.5541	.4799	.3918	.9685	.7908	.6849	.5592

TABLE 13

MONTE CARLO √MSEs FOR ESTIMATING L_{p^*}

MODEL: SKEWED LOGIT $\theta_1 = -1$ $\theta_2 = .7$

INITIAL SET 2

(136 discarded samples)

p^*	.5				.75			
N	10	15	20	30	10	15	20	30
RM-1	1.0013	1.0332	.9373	.8649	1.4682	1.5397	1.3767	1.3399
RM-6	1.0312	1.0820	1.0054	.9485	1.5099	1.6089	1.4741	1.4602
RM-36	1.0365	1.0906	1.0176	.9634	1.5172	1.6208	1.4913	1.4817
AN1-10	1.3032	1.0093	.9081	.7075	1.6796	1.4855	1.4012	1.2245
AN1-50	1.3260	1.0304	.9339	.7214	1.8162	1.6120	1.4432	1.2214
AN1-100	1.3578	1.0608	.9435	.7290	1.9127	1.6671	1.4272	1.2757
AN2-10	.8384	.7551	.5987	.4983	1.2177	1.0979	.8664	.7457
AN2-50	.9956	.8386	.7038	.5799	1.5615	1.2692	1.0709	.8861
AN2-100	1.1452	.9449	.7842	.7330	1.7349	1.3767	1.2120	1.0283
WU1-10	.7567	.4313	.3727	.3129	1.0227	.9412	.8874	.8657
WU1-50	.4824	.4173	.3615	.3071	.8263	.8335	.8215	.8021
WU1-100	.4925	.5349	.3663	.3110	.8293	.9466	.8188	.8020
WU2-10	.8675	.7738	.6353	.5434	1.2544	1.1140	.8953	.7566
WU2-50	.8946	.7624	.6591	.5617	1.2081	1.0430	.8637	.7562
WU2-100	.9739	.7667	.6861	.5604	1.3024	.9957	.8544	.7018
SAM-10	.5799	.5716	.4603	.3833	.9857	.8113	.7014	.6032
SAM-50	.5875	.5239	.4576	.3852	.8557	.7913	.6907	.5887
SAM-100	.5712	.5286	.4589	.3847	.8448	.7869	.6930	.5976
Asympt. RM	.6787	.5541	.4799	.3918	.9685	.7908	.6849	.5592

TABLE 14

MONTE CARLO √MSEs FOR ESTIMATING L_{p*}

MODEL: LOGLOG $\theta_1 = 0$ $\theta_2 = .5$

INITIAL SET 1

(144 discarded samples)

p^*	.5				.75			
N	10	15	20	30	10	15	20	30
RM-1	.9190	.9885	.9624	.9067	1.5460	1.5429	1.6181	1.5439
RM-6	.9526	1.0464	1.0390	1.0025	1.5916	1.6178	1.7216	1.6757
RM-36	.9585	1.0566	1.0528	1.0199	1.5994	1.6308	1.7397	1.6990
AN1-10	1.0087	.8072	.7437	.6114	1.5543	1.3606	1.3097	1.1300
AN1-50	1.0212	.8629	.7693	.6602	2.2065	1.6957	1.4913	1.1618
AN1-100	1.0953	.8960	.7739	.6672	2.1372	1.7189	1.4929	1.1573
AN2-10	.7598	.7289	.6348	.5444	1.2426	1.0586	.9326	.7143
AN2-50	.7735	.7047	.6286	.5620	1.2594	1.0121	.8547	.6711
AN2-100	.7933	.7056	.6298	.5607	1.3197	1.0239	.8655	.6769
WU1-10	.7983	.5804	.5226	.4312	1.8197	1.4150	1.5318	1.4156
WU1-50	.7112	.5788	.5313	.4311	1.0785	.8857	.9624	.7863
WU1-100	.7150	.5788	.5423	.4316	1.0788	.8994	.9858	.7745
WU2-10	.8002	.7600	.6433	.6050	1.2664	1.0651	.9243	.7059
WU2-50	.8649	.7592	.6732	.6176	.8756	.6777	.5783	.5190
WU2-100	.8441	.7351	.6879	.6152	.9246	.6581	.5829	.5183
SAM-10	.7191	.6703	.6131	.5301	.9303	.6960	.6321	.4946
SAM-50	.7315	.6694	.6171	.5204	.9173	.7067	.6299	.5010
SAM-100	.7326	.6787	.6143	.5182	.9172	.7075	.6301	.4993
Asympt. RM	.8390	.6850	.5932	.4844	.7540	.6157	.5332	.4353

TABLE 15

MONTE CARLO √MSEs FOR ESTIMATING L_{p^*}

MODEL: LOGLOG $\theta_1 = 0$ $\theta_2 = .5$

INITIAL SET 2

(157 discarded samples)

p^*	.5				.75			
N	10	15	20	30	10	15	20	30
RM-1	.7921	.7330	.7789	.7074	1.3210	1.3215	1.3121	1.2341
RM-6	.7052	.6152	.5842	.4969	1.1567	1.0553	.9405	.7891
RM-36	1.0948	1.0186	.9063	.7814	1.1586	1.0124	.8459	.7148
AN1-10	.8878	.7117	.6467	.5485	1.4560	1.3061	1.2510	1.1514
AN1-50	.9047	.7588	.6794	.5928	1.6918	1.4569	1.2483	1.1305
AN1-100	1.0124	.7917	.7080	.5957	1.8664	1.4939	1.2726	1.1464
AN2-10	.6975	.6086	.5852	.5058	1.1160	.9815	.8429	.6811
AN2-50	.8725	.7367	.7105	.6285	1.2706	1.0673	.9335	.7318
AN2-100	1.0576	.8493	.7890	.7031	1.5837	1.2899	1.1069	.8596
WU1-10	.7029	.5318	.4431	.3760	.9573	.9214	.7994	.7555
WU1-50	.6680	.5985	.5650	.6278	.9098	.8807	.8957	.9746
WU1-100	.7765	.6028	.5726	.5201	1.0111	.8875	.9002	.8788
WU2-10	.7600	.7136	.6807	.6131	1.1344	.9944	.8187	.6335
WU2-50	1.0386	.8117	.7560	.6526	.9885	.7453	.6047	.5698
WU2-100	1.0193	.8425	.7669	.6524	1.1010	.8634	.6767	.5534
SAM-10	.7091	.6341	.5344	.4541	.7755	.7251	.5897	.4506
SAM-50	.7265	.6387	.5777	.4767	.8548	.8019	.6178	.4768
SAM-100	.7633	.6599	.5561	.5022	.8257	.7054	.6071	.4777
Asympt. RM	.8390	.6850	.5932	.4844	.7540	.6157	.5332	.4353

levels ($x_{1,1}$ and $x_{1,2}$) are needed for these procedures. For AN1 and WU1, a single RM procedure with $p = .5$ generates the initial design levels, and a single starting level (x_1) is required. Three different sets of starting levels, $(L_{.1}, L_{.4}, L_{.7})$, $(L_{.5}, L_{.813}, L_{.95})$, $(L_{.3}, L_{.35}, L_{.4})$, are used in this study. The first and third values of each triple are the starting levels for SAM, AN2, and WU2. The middle value is the starting level for AN1 and WU1.

The updates at which the six procedures switch from the initial RM procedure to their own updating rules are given below.

AN1 -- the first $n \geq 10$ such that b_n (6) is nonzero

AN2 -- each of the two independent Anbar processes is started separately at the first $n \geq 5$ such that b_n is nonzero.

WU1 -- the first $n \geq 10$ such that the MLEs exist

WU2 -- each of the two independent Wu procedures is started separately at the first $n \geq 5$ such that the MLEs exist

SAM -- the first $n \geq 5$ such that the MLEs from the two combined independent RM initial procedures exist

An upper bound of $\delta_2 = 100$ and a lower bound of $\delta_1 = .05$ is used for all procedures. The MSE, the average of $(\hat{L}_{p*} - L_{p*})^2$ over the 500 samples, is again used to compare the procedures.

Results of Initial Procedure 2

Tables 16 - 24 contain the MSEs for the logit and loglog models at each of the three starting levels. The MSEs of both $\hat{L}_{.5}$ and $\hat{L}_{.75}$ are given in Tables 16 - 21. The MSEs of $\hat{L}_{.25}$ were also obtained in each situation, and are presented seperately in Tables 22 - 24. The results for $\hat{L}_{.25}$ and $\hat{L}_{.75}$ are fairly similar. Thus, the conclusions in the discussion below for $L_{.75}$ hold also for $L_{.25}$, except when mentioned specifically.

For estimating $\hat{L}_{.75}$, SAM consistently produced the lowest MSEs of any procedure. Note that SAM was superior for both the logit and loglog models. An exception occurred with starting values $(L_{.3}, L_{.35}, L_{.4})$. With $n \cdot A_n = 1$, the RM-1 procedure produced the lowest MSEs of any procedure. However, for estimating $L_{.25}$ in this situation, the RM-1 produced larger MSEs than SAM, except when n = 10.

The comparisons between the two Wu procedures and between the two Anbar procedures differed from initial procedure 1. Using initial procedure 2, WU2 proved to be superior to WU1 for estimating $L_{.75}$. In initial procedure 1, the initial design levels for the procedures are evenly spread throughout the distribution. With initial procedure 2, however, procedures such as WU1 and AN1 may have their initial design levels in one section of the distribution. This would make the procedure less efficient in estimating $L_{.75}$ and $L_{.25}$ for small sample sizes. However, the MSEs from AN2 for estimating $L_{.75}$ were not always better than AN1.

For estimating $L_{.5}$, no one procedure performed the best in all situations. In various situations, estimates from WU2, AN2, and SAM all produced the lowest MSEs of any procedure. In most cases, these three were superior to AN2 WU2 and RM. Regardless of the model, WU1 produced the lowest MSEs for $\hat{L}_{.5}$ when $n \cdot A_n = 36$. Using the larger step size and spreading out the initial design levels appeared to help the WU1 procedure. For starting values $(L_{.5}, L_{.813}, L_{.95})$, SAM generally produced the lowest MSEs when $n \cdot A_n = 1$, while AN1 produced the lowest when $n \cdot A_n = 6$. For the other starting levels with $n \cdot A_n \neq 36$, no clear pattern emerged.

The RM procedure worked well only in specific situations. With starting values $(L_{.1}\ L_{.7},\ L_{.4})$ and $n \cdot A_n =$ 1, the RM procedure performed very well. Using $n \cdot A_n = 1$ does not allow the design levels to change very quickly. Thus, when the starting levels are close to the true values, the RM-1 procedure performs well. When the initial starting levels are not good, however, RM-1 did not perform well. Using the near optimal RM-6 was preferable to RM-1 in these situations. When n = 10, the RM procedures occasionally produced $\hat{L}_{.5}$ MSEs of similar size as SAM and WU1. This could be a result of the instability of parameter estimates for very small sample sizes.

For a given model, as n increased to 30, SAM produced fairly constant MSEs. That is, the MSEs produced by SAM were very similar regardless of the starting levels and values of $n \cdot A_n$. For the logit model with n = 30, SAM produced MSEs for $\hat{L}_{.5}$ and $\hat{L}_{.75}$ near .33 and .45, respectively. Using initial procedure 1, the MSEs for SAM in this situation were near .32 and .44.

The time at which the procedures switched from the initial RM procedure to their own updating rules is a random variable. For SAM, the average number of updates until MLEs existed ranged from 5.23 to 8.51 . The smaller averages occurred when the near optimal constant $n \cdot A_n = 6$ was used in the initial RM design. The larger averages resulted when $n \cdot A_n = 1$ and the loglog model was used to generate the binary responses.

Analysis of the MSEs

In this section, the differences between SAM's and Wu's MSEs (Tables 8 - 24) are studied. Least significant differences for the MSEs produced by SAM and WU1 are derived.

For SAM, using the first order asymptotic results of Chapter III (equation (29)),

$$\hat{L}_p \sim N\left(L_p,\ c_L^{-2}p(1-p)\ /\ n(2c_L^{-1}M'(L_p)\ -\ 1)\right) \quad \text{and} \tag{58}$$

$$\hat{L}_{1-p} \sim N\left(L_{1-p},\ c_L^{-2}p(1-p)\ /\ n(2c_L^{-1}M'(L_{1-p})\ -\ 1)\right),$$

TABLE 16

MONTE CARLO √MSEs FOR ESTIMATING L_{p*}

MODEL: LOGIT $\theta_1 = 0$ $\theta_2 = 1$

STARTING LEVELS (.5, .813, .95)

	$p^* = .5$				$p^* = .75$			
N	10	15	20	30	10	15	20	30
$n \cdot A_n = 1$								
RM	.9676	.8996	.8895	.8380	.8581	.7825	.7633	.6978
AN1	1.2232	.9122	.7995	.6128	1.6577	1.4059	1.2209	1.0132
AN2	.9569	.8482	.8016	.7524	1.2411	1.1660	1.1057	.9083
WU1	.6897	.5711	.5312	.4417	1.5677	1.3423	1.2834	1.1706
WU2	.9212	.8369	.8013	.6949	.9515	.8141	.7947	.6172
SAM	.6324	.5005	.4324	.3242	.8276	.6915	.5772	.4533
$n \cdot A_n = 6$								
RM	.6727	.5246	.4707	.3947	1.0237	.8478	.7060	.5917
AN1	.5733	.4508	.3831	.3057	.9178	.7306	.6733	.6326
AN2	.7406	.6325	.5747	.4939	1.1699	1.0023	.8787	.7231
WU1	.5761	.4529	.3922	.3148	1.0027	.8724	.7887	.7559
WU2	.7237	.6035	.5332	.4585	1.1134	.9238	.7536	.6086
SAM	.6692	.4803	.4178	.3314	.9190	.6912	.5807	.4781
$n \cdot A_n = 36$								
RM	1.2003	.9396	.7655	.5803	1.5008	1.2514	1.0327	.7801
AN1	.7784	.5207	.4161	.3129	1.1845	1.0191	.9334	.9078
AN2	.9666	.7456	.6461	.4749	1.2344	.9836	.8087	.6276
WU1	.6731	.4992	.4139	.3877	1.6660	.8786	.7308	.6971
WU2	1.0500	.7066	.5587	.3890	1.2568	.8920	.6365	.4826
SAM	.8293	.5383	.4462	.3471	.9893	.6643	.5252	.4553

TABLE 17

MONTE CARLO $\sqrt{MSE}$s FOR ESTIMATING L_{p^*}

MODEL: LOGIT $\theta_1 = 0$ $\theta_2 = 1$

STARTING LEVELS (.1, .4, .7)

	$p^* = .5$				$p^* = .75$			
N	10	15	20	30	10	15	20	30
$n \cdot A_n = 1$								
RM	.5163	.4954	.4561	.4466	.6219	.6239	.5563	.5469
AN1	.5815	.5453	.5050	.4347	.8882	1.0334	.9145	.8282
AN2	.5671	.5560	.5438	.5715	.6279	.6517	.6079	.6410
WU1	.4543	.4231	.4019	.3522	1.5726	1.0297	1.0358	.8926
WU2	.5331	.5090	.4705	.4636	.6074	.6063	.5412	.5535
SAM	.5627	.4811	.4010	.3291	.6242	.6063	.4896	.4256
$n \cdot A_n = 6$								
RM	.6292	.5122	.4715	.3726	.8112	.7003	.5910	.4816
AN1	.5569	.4599	.4200	.3189	.7678	.6239	.6251	.5659
AN2	.6967	.6327	.5659	.4754	.8918	.8252	.7230	.6098
WU1	.5518	.4689	.4052	.3448	1.1704	.9188	.9048	.8500
WU2	.6870	.5611	.5442	.4545	.7951	.6953	.6455	.5645
SAM	.5894	.4553	.4018	.3199	.7223	.6102	.5193	.4241
$n \cdot A_n = 36$								
RM	1.1532	.9262	.7626	.5725	1.4708	1.2369	1.0169	.7939
AN1	.7902	.5182	.4147	.3083	1.2395	1.0375	.9738	.8981
AN2	.9949	.7358	.6427	.4741	1.3887	1.0653	.8582	.6432
WU1	.6216	.4539	.3778	.3179	1.5324	.8841	.6584	.5803
WU2	.9661	.6710	.5284	.3868	1.2915	.8657	.6749	.5225
SAM	.8232	.5763	.4468	.3470	1.0027	.7068	.5555	.4735

TABLE 18

MONTE CARLO $\sqrt{MSE}$s FOR ESTIMATING L_{p^*}

MODEL: LOGIT $\theta_1 = 0$ $\theta_2 = 1$

STARTING LEVELS (.3, .35, .4)

	$p^* = .5$				$p^* = .75$			
N	10	15	20	30	10	15	20	30
$n \cdot A_n = 1$								
RM	.4299	.4006	.3810	.3279	.4114	.3850	.3705	.3502
AN1	.6625	.6380	.5806	.4893	.8330	1.3193	.8740	.8202
AN2	.8924	.8927	.8735	.7830	.7929	.8123	.8709	.7892
WU1	.5078	.4661	.3888	.3661	1.4422	1.1002	1.0522	.9294
WU2	.6354	.4667	.4730	.3900	.6171	.4872	.5413	.4888
SAM	.5766	.4162	.3881	.3198	.6918	.5550	.5046	.4354
$n \cdot A_n = 6$								
RM	.6457	.5022	.4714	.3877	.9415	.7864	.6576	.5536
AN1	.5724	.4712	.4148	.3305	.8016	.6324	.5923	.5384
AN2	.7600	.6037	.5906	.4871	1.1323	.9677	.9160	.7309
WU1	.5786	.4857	.4098	.3448	1.0900	.8797	.8431	.7576
WU2	.7774	.5921	.5455	.4539	1.0440	.8546	.7053	.5903
SAM	.6318	.4869	.4262	.3244	.7354	.6229	.5697	.4277
$n \cdot A_n = 36$								
RM	1.1413	.9157	.7740	.5629	1.4851	1.2462	1.0373	.7768
AN1	.7944	.5212	.4148	.3075	1.2405	1.0459	.9715	.8935
AN2	.8638	.6345	.5727	.4456	1.2799	.9612	.7894	.6069
WU1	.6081	.4581	.3740	.3130	1.4300	.8428	.6436	.5921
WU2	.9671	.6616	.5011	.3642	1.2215	.8660	.6461	.4703
SAM	.7726	.5237	.4489	.3316	.9005	.6536	.5599	.4511

TABLE 19

MONTE CARLO √MSEs FOR ESTIMATING L_{p*}

MODEL: LOGLOG $\theta_1 = 0$ $\theta_2 = .5$

STARTING LEVELS (.5, .813, .95)

	$p^* = .5$				$p^* = .75$			
N	10	15	20	30	10	15	20	30
$n \cdot A_n = 1$								
RM	.9484	.8739	.8633	.7965	1.4461	1.3541	1.3346	1.2442
AN1	1.4208	1.1946	1.0672	.8321	2.2488	2.0361	1.7992	1.4677
AN2	1.0767	1.0077	.9729	.9226	1.8137	1.7403	1.6522	1.4014
WU1	1.0810	.8048	.7602	.6245	2.3629	2.1073	2.0123	1.8210
WU2	.9614	.8348	.7609	.6815	1.5535	1.3540	1.2354	1.1070
SAM	.8685	.6750	.5911	.5007	1.3554	1.0951	.9636	.7929
$n \cdot A_n = 6$								
RM	.9016	.7892	.7258	.6586	1.1937	1.0962	.9690	.8589
AN1	.8201	.6573	.5779	.4533	1.2772	1.1285	1.0329	.9662
AN2	1.0813	.9433	.8901	.7731	1.5134	1.3727	1.2857	1.0807
WU1	.8643	.6863	.5902	.4993	1.5742	1.3641	1.3061	1.2363
WU2	1.0353	.9105	.8634	.7751	1.4433	1.2668	1.1562	.9929
SAM	.9152	.7495	.6352	.5567	1.3553	1.1099	.9378	.8135
$n \cdot A_n = 36$								
RM	1.2669	1.1305	.9521	.8245	1.8381	1.5493	1.3352	1.1209
AN1	1.0013	.7514	.5942	.4611	1.5146	1.3379	1.1676	1.1028
AN2	1.2060	1.0429	.9738	.8021	1.7766	1.5001	1.3353	1.1040
WU1	.8641	.6530	.5465	.4355	1.8460	1.3205	1.2011	1.0561
WU2	1.1235	.9286	.7527	.6275	1.5240	1.1260	.9748	.8130
SAM	1.0561	.8667	.6991	.5488	1.3163	1.0588	.8990	.7575

TABLE 20

MONTE CARLO √MSEs FOR ESTIMATING L_{p^*}

MODEL: LOGLOG $\theta_1 = 0$ $\theta_2 = .5$

STARTING LEVELS (.1, .4, .7)

	$p^* = .5$				$p^* = .75$			
N	10	15	20	30	10	15	20	30
$n \cdot A_n = 1$								
RM	1.1269	1.1035	1.0570	1.0528	1.2906	1.2820	1.2131	1.1982
AN1	.7483	.6790	.6482	.5567	1.3702	1.4685	1.6135	1.3051
AN2	1.1559	1.1328	1.0576	.9977	1.2884	1.2933	1.1920	1.1793
WU1	.5964	.5849	.5265	.5069	2.0399	1.6280	1.6166	1.4759
WU2	1.1342	1.1008	1.0493	1.0179	1.2750	1.2631	1.1848	1.1648
SAM	.9890	.8397	.7012	.5785	1.1509	1.0287	.8383	.7088
$n \cdot A_n = 6$								
RM	.8992	.7984	.6936	.6435	1.2227	1.1696	.9687	.8828
AN1	.7933	.6855	.5913	.4875	1.2080	1.0712	.9930	.9099
AN2	1.0324	.9294	.8061	.7582	1.3161	1.2959	1.1104	1.0582
WU1	.8312	.6819	.6094	.5148	1.6961	1.4106	1.4350	1.2961
WU2	.9464	.8494	.7348	.7022	1.1634	1.1029	.9267	.9046
SAM	.8583	.7611	.6556	.5456	1.1125	.9875	.8297	.7126
$n \cdot A_n = 36$								
RM	1.2803	1.1148	.9804	.8161	1.7679	1.5598	1.3536	1.1340
AN1	.9916	.7451	.5985	.4486	1.7171	1.4616	1.3160	1.2263
AN2	1.2637	1.0472	.9719	.7738	1.7385	1.4355	1.2739	1.0477
WU1	.8169	.6027	.5293	.4264	1.6243	1.0958	1.0444	.9988
WU2	1.1372	.9064	.7691	.6687	1.5968	1.2062	1.0556	.8668
SAM	1.1669	.8888	.7154	.5697	1.3360	1.0808	.9337	.7362

TABLE 21

MONTE CARLO √MSEs FOR ESTIMATING L_{p*}

MODEL: LOGLOG $\theta_1 = 0$ $\theta_2 = .5$

STARTING LEVELS (.3, .35, .4)

	$p^* = .5$				$p^* = .75$			
N	10	15	20	30	10	15	20	30
$n \cdot A_n = 1$								
RM	.7066	.6833	.6480	.6400	.5604	.5240	.5227	.5045
AN1	.9196	.8516	.7718	.6520	1.2484	2.5337	1.5865	1.2384
AN2	1.5434	1.3937	1.3246	1.0978	1.1141	1.1135	1.1726	1.0704
WU1	.7236	.7069	.6443	.5909	2.0421	1.6148	1.5053	1.3977
WU2	1.1139	.7228	.6440	.5380	.8997	.7083	.7192	.6858
SAM	.8331	.5928	.5743	.5047	1.0731	.8258	.7945	.7257
$n \cdot A_n = 6$								
RM	.9096	.7821	.7477	.6737	1.3688	1.2190	1.0873	.9375
AN1	.8075	.6567	.5939	.4658	1.2273	1.0416	1.0339	.9204
AN2	1.1536	.9696	.9632	.7809	1.5869	1.4481	1.3994	1.1659
WU1	.8108	.6775	.6014	.4835	1.6731	1.4099	1.4068	1.2497
WU2	1.0434	.8632	.8473	.7221	1.5264	1.3526	1.2299	1.0333
SAM	.8823	.6973	.6077	.5097	1.1967	.9610	.8746	.7149
$n \cdot A_n = 36$								
RM	1.2981	1.1426	.9788	.8051	1.8149	1.5441	1.3558	1.1161
AN1	.9954	.7438	.6008	.4506	1.7321	1.4625	1.3255	1.2449
AN2	1.4404	1.1207	1.0144	.7557	1.7413	1.4326	1.2691	1.0521
WU1	.8217	.6124	.5255	.4247	1.6406	1.0710	1.0124	.9805
WU2	1.1856	.9032	.7613	.6485	1.4704	1.1196	.9642	.8224
SAM	.9955	.7727	.6775	.5394	1.1724	1.0337	.8909	.7157

TABLE 22

MONTE CARLO √MSE FOR ESTIMATING $L_{.25}$

STARTING LEVELS: (.5, .813, .95)

	LOGIT $\theta_1 = 0$, $\theta_2 = 1$				LOGLOG $\theta_1 = 0$, $\theta_2 = .5$			
N	10	15	20	30	10	15	20	30
$n \cdot A_n = 1$								
RM	1.1268	1.0689	1.0570	1.0237	.8449	.7869	.7807	.7418
AN1	1.2547	.9646	.8018	.6215	1.4685	1.2786	1.0222	.7346
AN2	1.0963	1.0084	.9638	.9844	.8012	.7076	.7144	.8103
WU1	.7273	.5978	.6061	.5866	1.1015	.7542	.8273	.6181
WU2	1.0993	1.0413	1.0008	.9391	.8010	.7367	.7045	.6970
SAM	.8695	.7264	.6680	.4952	1.0379	.8329	.7470	.5866
$n \cdot A_n = 6$								
RM	.8190	.6270	.6032	.4866	1.0579	.8061	.7647	.6974
AN1	.7373	.6401	.6411	.5593	1.0493	.8898	.8281	.7192
AN2	.8917	.7746	.7397	.6267	1.1564	.9169	.8865	.7614
WU1	.9926	.8956	.7881	.7185	1.4348	1.2682	1.1313	1.0266
WU2	.7600	.6492	.6215	.5589	1.0584	.8548	.8439	.7870
SAM	.7026	.5911	.5300	.4337	.8464	.7386	.6194	.5352
$n \cdot A_n = 36$								
RM	1.5931	1.1894	1.0162	.7706	1.3626	1.1545	.9582	.7907
AN1	1.1791	.9875	.9074	.8523	1.6177	1.4814	1.3738	1.2521
AN2	1.4456	1.0889	.9323	.7187	1.0717	.8553	.7436	.6282
WU1	1.6580	.8726	.7131	.6264	2.2801	1.5156	1.2455	.9331
WU2	1.4050	.8764	.7560	.5259	1.3337	1.0901	.8080	.6089
SAM	1.0156	.6835	.6021	.4513	1.0866	.8932	.7207	.5488

TABLE 23

MONTE CARLO $\sqrt{MSE}$ FOR ESTIMATING $L_{.25}$

STARTING LEVELS: (.1, .4, .7)

	LOGIT $\theta_1 = 0$, $\theta_2 = 1$				LOGLOG $\theta_1 = 0$, $\theta_2 = .5$			
N	10	15	20	30	10	15	20	30
$n \cdot A_n = 1$								
RM	.4990	.4716	.4556	.4330	.6734	.6451	.6208	.6117
AN1	1.1866	1.2198	1.0902	1.0269	1.6242	1.6501	1.6795	1.4218
AN2	.7104	.6665	.7305	.7666	1.0572	.9666	1.0215	1.0289
WU1	1.6893	1.3160	1.2499	1.1337	2.6911	2.0854	1.9594	1.7532
WU2	.5907	.5613	.5319	.5160	.8121	.7733	.6812	.6554
SAM	.7154	.6334	.5300	.4343	.9571	.7663	.6217	.5121
$n \cdot A_n = 6$								
RM	.8982	.7489	.7003	.5355	1.0680	.8418	.7313	.5616
AN1	.7503	.6545	.5759	.5139	1.1948	1.0420	.8513	.7795
AN2	1.0477	.9669	.8409	.7257	1.3816	1.1839	1.0031	.8193
WU1	1.1836	.9882	.9081	.8755	1.8005	1.4046	1.3015	1.1618
WU2	.9614	.7967	.7586	.5800	1.2031	.9839	.8657	.6567
SAM	.7817	.6553	.5855	.4632	.9384	.8277	.7241	.5622
$n \cdot A_n = 36$								
RM	1.5564	1.1777	1.0117	.7597	1.4183	1.1059	.9750	.7613
AN1	1.1645	.9750	.9544	.8536	1.7061	1.5724	1.4786	1.3964
AN2	1.2994	.9565	.8053	.6356	1.4965	1.2582	1.0655	.8394
WU1	1.5460	.9169	.7175	.6231	2.0142	1.2082	.9724	.8708
WU2	1.2495	.8342	.6738	.5059	1.0924	.8423	.7020	.5747
SAM	1.0016	.7201	.5741	.4601	1.2227	.9092	.7121	.5665

TABLE 24

MONTE CARLO √MSE FOR ESTIMATING $L_{.25}$

STARTING LEVELS: (.3, .35, .4)

	LOGIT $\theta_1 = 0,\ \theta_2 = 1$				LOGLOG $\theta_1 = 0,\ \theta_2 = .5$			
N	10	15	20	30	10	15	20	30
$n \cdot A_n = 1$								
RM	.8961	.8073	.7755	.7419	1.3309	1.2376	1.1893	1.1457
AN1	1.2605	1.5418	1.1532	1.0771	1.7769	2.6936	1.7967	1.4782
AN2	1.3334	1.2959	1.2075	1.1166	2.4649	2.1663	1.9818	1.6276
WU1	1.7436	1.4712	1.3369	1.2133	2.9345	2.3625	2.0877	1.8928
WU2	1.0369	.7854	.7339	.5976	1.8223	1.1401	.9773	.7720
SAM	.9578	.7452	.6471	.5355	1.4227	.9218	.8015	.6704
$n \cdot A_n = 6$								
RM	.8972	.7434	.6818	.5138	.8972	.7434	.6818	.5138
AN1	.8028	.6693	.5874	.5200	.8028	.6693	.5874	.5200
AN2	1.0591	.8760	.7871	.6765	1.0591	.8760	.7871	.6765
WU1	1.1488	.9938	.8706	.8114	1.1488	.9938	.8706	.8114
WU2	1.0273	.7917	.7390	.5709	1.0273	.7917	.7390	.5709
SAM	.8392	.6699	.5666	.4636	.8392	.6699	.5666	.4636
$n \cdot A_n = 36$								
RM	1.5148	1.1366	1.0101	.7570	1.5148	1.1366	1.0101	.7570
AN1	1.1624	.9773	.9501	.8451	1.1624	.9773	.9501	.8451
AN2	1.1888	.9484	.7961	.6276	1.1888	.9484	.7961	.6276
WU1	1.4624	.8916	.7172	.6395	1.4624	.8916	.7172	.6395
WU2	1.2581	.8275	.6508	.4976	1.2581	.8275	.6508	.4976
SAM	1.0023	.7082	.6004	.4595	1.0023	.7082	.6004	.4595

where $c_L = 2p(1-p)\ln[(1-p)/p]/(L_{1-p} - L_p)$. When the true expectation, M(x), is the two parameter logit model, (58) reduces to

$$\hat{L}_p \sim N\left(L_p, \{n\theta_2^2 p(1-p)\}^{-1}\right) \quad \text{and} \tag{59}$$

$$\hat{L}_{1-p} \sim N\left(L_{1-p}, \{n\theta_2^2 p(1-p)\}^{-1}\right).$$

Thus, using (38) and (58), when M(x) is the two parameter logit model,

$$\hat{L}_{p*} \sim N\left(L_{p*}, \{n\theta_2^2 p(1-p)\}^{-1}\{1-2r+2r^2\} \right), \tag{60}$$

where $r = (1/2) + \{\ln[(1-p^*)/p^*] \,/\, 2\cdot\ln[(1-p)/p]\}$.

The square of the values in Tables 8 - 24 are averages of n^* ($n^* = n'$ for initial procedure 1 and $n^* = 500$ for initial procedure 2) individual mean squares, $(\hat{L}_{p*} - L_{p*})^2$. Define the random variable $S = (\hat{L}_{p*} - L_{p*})^2$. Using (60), asymptotically, $S^* = S\cdot\{1-2r+2r^2\}^{-1}\cdot\{n\theta_2^2 p(1-p)\} \sim \chi_1^2$. This implies the asymptotic variance of S is $2\cdot(1-2r+2r^2)^2/(n\theta_2^2 p(1-p))^2$. By the central limit theorem, an average of n^* of these random variables is approximately normal with a variance of $2\cdot(1-2r+r^2)^2/\, n^*(n\theta_2^2 p(1-p))^2$.

The same approach can be followed for $\hat{L}_{.5}$ from WU1, using the asymptotic equivalence of Wu's procedure and the optimal RM process. Tables 25 and 26 give the asymptotic standard errors of SAM's and WU1's MSEs for $n^* = 365$ and $n^* = 500$, respectively, when M(x) is given by the two parameter logit model.

TABLE 25

ASYMPTOTIC (MSE) STANDARD ERRORS ($n^* = 365$)

	p^*	n = 10	15	20	30
SAM	.5	.0231	.0154	.0116	.0077
	.75	.0377	.0251	.0188	.0126
WU1	.5	.0148	.0099	.0074	.0049

TABLE 26

ASYMPTOTIC (MSE) STANDARD ERRORS ($n^* = 500$)

	p^*	n = 10	15	20	30
SAM	.5	.0198	.0132	.0099	.0066
	.75	.0322	.0215	.0161	.0107
WU1	.5	.0126	.0084	.0063	.0042

These values have been developed assuming the true expectation is the two parameter logit. Thus, they are appropriate for comparing SAM and WU1's MSEs in Tables 8, 9, 16, 17 and 18. If M(x) differs from the two parameter logit, then $\hat{L}_{p^*}$ may be biased. Moser and Fei (1989b) provide a detailed discussion of the biases and MSEs for a two dimensional Robbins Monro process.

The above results are based upon asymptotic theory. However, small to medium sample sizes were used in the simulation study. A second approach to estimating the variance of the MSEs is to use the simulation results.

During the simulation study, the sample standard deviation of $S = (\hat{L}_{p*} - L_{p*})^2$ was calculated for each procedure. Table 27 contains the standard errors of S from initial procedure 1 when the logit model was used to generate the binary responses.

TABLE 27

SIMULATION STANDARD ERRORS ($n^* = 365$)

Procedure	p^*	n = 10	15	20	30
SAM	.5	.0247	.0178	.0143	.0084
	.75	.0532	.0348	.0253	.0137
WU1	.5	.0230	.0194	.0112	.0065
	.75	.0764	.0618	.0462	.0268

In every case, the difference between the asymptotic (Table 25) and simulation standard errors (Table 27) is smaller for SAM than for WU1. Using the simulation standard erros from Table 27, denoted by $s_{SAM} / \sqrt{n^*}$ and $s_{WU1} / \sqrt{n^*}$, the least significant difference between MSEs from SAM and WU1 can be constructed. For example, the least significant differences for the MSEs, $z_{\alpha/2} \cdot \left(\frac{1}{n^*} \cdot (s^2_{SAM} + s^2_{WU1})\right)^{1/2}$, are given in Table 28.

TABLE 28

LEAST SIGNIFICANT DIFFERENCE FOR SAM AND WU1 MSEs ($n^* = 365$, $\alpha = .05$)

	n			
p	10	15	20	30
.5	.0662	.0516	.0356	.0208
.75	.1825	.1390	.1032	.0600

The least significant differences of Table 28 are now used to compare the MSEs produced by SAM and WU1 in Tables 8 and 9. Tables 8 and 9 present the MSEs from each procedure for four sample sizes and three upper bounds. Thus, in each table, there are 12 comparisons between SAM and WU1's MSEs for $\hat{L}_{.5}$ and 12 for $\hat{L}_{.75}$. (For each sample size (n=10, 15, 20, 30), the MSE of SAM-10 is compared with the MSE of WU1-10, SAM-50 is compared with WU1-50, and SAM-100 is compared with WU1-100) In Table 29, the results of these comparisons over both tables are presented. The number of times that each procedure produced a MSE significantly lower ($\alpha = .05$) than the other procedure is given for each sample size. Note that for each sample size, there are a total of six $\hat{L}_{.5}$ comparisons and six $\hat{L}_{.75}$ comparisons (three from each table).

TABLE 29

NUMBER OF SIGNIFICANT DIFFERENCES

	n = 10	15	20	30
$\hat{L}_{.5}$				
WU1	2	5	4	0
SAM	1	0	0	3
Neither	3	1	2	3
$\hat{L}_{.75}$				
WU1	0	0	0	0
SAM	1	0	6	6
Neither	5	6	0	0

It is clear that SAM produces significantly ($\alpha = .05$) lower $\hat{L}_{.75}$ MSEs for the larger sample sizes (n = 20, 30). For estimating $L_{.5}$, WU1 produces significantly lower MSEs when n = 15, 20. However, WU1's advantage did not hold for n = 10, 30.

CHAPTER V

CONCLUSIONS

SAM provides a new approach to estimating multiple roots of an expectation function. A parametric model is used to produce SAM's estimators, although the true expectation is assumed unknown. To produce estimators for binary data, SAM, with the two parameter logit model, is recommended. A first order approximation to the logit version of SAM was shown to be asymptotically equivalent to a two dimensional Robbins-Monro process. Under certain restrictions, SAM's estimators were proven to be consistent. In the binary data simulation study, SAM performed well compared to other sequential approximation methods. SAM performed particularly well when estimating multiple roots. SAM was relatively unaffected by the choice of bounds on the step size and designs used to generate the initial levels.

Criteria for selecting the values of $p_1,\ldots,p_k$ to use in SAM were presented in Chapter II. For the two parameter logit model, using $p_1 = .2$ and $p_2 = .8$ is recommended. The pair $(p_1,p_2) = (.2,.8)$ was found to be either optimal (in the minimum average sense) or near optimal in the situations discussed in Chapter II. Also, using $(p_1,p_2) = (.2,.8)$, SAM performed well for estimating multiple roots in the simulation study.

In order to use SAM's updating rule, MLEs must exist. Therefore, some procedure other than SAM must be used for the initial updates. Two methods for obtaining the initial design levels have been presented in this paper. The first approach is to select an initial set of design levels and observe a fixed number of samples at these levels (initial procedure 1). If MLEs do not exist after these initial observations, then more design levels would have to be selected. A second approach is to use a RM procedure, then switch to SAM when the MLEs exist. If little information about the location of the roots of M(x) is available prior to the experiment, then the second approach is recommended.

BIBLIOGRAPHY

Anbar, D. (1977), "A Stochastic Newton-Raphson Method," *Journal of Statistical Planning and Inference*, 2, 153 - 164.

Anton, H. (1980), Calculus With Analytic Geometry, New York: Wiley.

Blum, J. R. (1954), "Approximation Methods Which Converge With Probability One," *Annals of Mathematical Statistics*, 25, 382 - 386.

Chung, K. L. (1954), "On a Stochastic Approximation Method," *Annals of Mathematical Statistics*, 25, 463 - 483.

Cochran, W. G. and Davis, M (1965), "The Robbins-Monro Method for Estimating the Median Lethel Dose," *Journal of the Royal Statistical Society, Series B*, 27, 28 - 44.

Cox, D. R. (1975), "Partial Likelihood," *Biometika*, 62, 269 - 276.

Crowder, M. J. (1975), "Maximum Likelihood Estimation for Dependent Observations," *Journal of the Royal Statistical Society, Series B*, 38, 45 - 53.

Davis, M. (1971), "Comparisons of Sequential Bioassays in Small Samples," *Journal of the Royal Statistical Society, Series B*, 33, 78 - 87.

Dixon, W. J. and Mood, A. M. (1948), "A Method for Obtaining and Analyzing Sensitivity Data," *Journal of the American Statistical Association*, 43, 109 - 126.

Dubins, L. E. and Freeman, P. R. (1965), "A Sharper Form of the Borel-Cantelli Lemma and the Strong Law," *Annals of Mathematical Statistics*, 36, 800 - 807.

Fei, W. C. (1989), "A New Method for the Robbins-Monro Stochastic Approximation Procedure," unpublished Ph.D. thesis, Oklahoma State University, May, 1989.

Freeman, P. R. (1970), "Optimal Bayesian Sequential Estimation of the Median," *Biometrika*, 57, 79 - 89.

Lord, F. M. (1971), "Tailored Testing, An Application of Stochastic Approximation," *Journal of the American Statistical Association*, 66, 707 - 711.

Langlie, H. J. (1962), "A Reliability Test Method for One Shot Items," U-1792, Ford Motor Co., Newport Beach, California.

Lai, T. L. and Wei, C. Z. (1982), "Least Squares Estimates in Stochastic Regression Models with Applications to Identification and Control of Dynamic Systems," *Annals of Statistics*, 10, 154-166.

McCullagh, P. (1981), Discussion of "Randomized Allocation of Treatments in Sequential Experiments," by Bather, *Journal of the Royal Statistical Society, Series B*, 43, 286 - 287.

Maron, M. J. (1982), Numerical Analysis, A Practical Approach, New York: Macmillan.

Mood, A. M., Graybill, F. A. and Boes D. C. (1974), Introduction to the Theory of Statistics, McGraw-Hill: Tokyo.

Moser, B. K. and Fei, W. C. (1989a), "A New Adaptive Robbins-Monro Procedure," submitted to *Metrika*, Nov., 1989.

Moser, B. K. and Fei, W. C. (1989b), "Bias and Optimal Root Selection for a New Adaptive Robbins-Monro Procedure," submitted to *Metrika*, Nov., 1989.

Neyer, B. T. (1989), "More Efficient Sensitivity Testing," EGG Mound Applied Technologies, Miamisburg, Ohio.

Robbins, H. and Monro, S. (1951), "A Stochastic Approximation Method," *Annals of Mathematical Statistics*, 29, 400 - 407.

Sacks, J. (1958), "Asymptotic Distribution of Stochastic Approximation Procedures," *Annals of Mathematical Statistics*, 29, 373 - 405.

SAS Institute Inc., (1985), SAS User's Guide: Basics, Version 5 Edition, Cary N.C..

Serfling, R. J. (1980), Approximation Theorems of Mathematical Statistics, New York: Wiley.

Silvapulle, M. J. (1981), "On the Existence of Maximum Likelihood Estimators for the Binomial Response Model," *Journal of the Royal Statistical Society, Series B*, 43, 310 - 313.

Venter, J. H. (1967), "An Extension of the Robbins-Monro Procedure," *Annals of Mathematical Statistics*, 38, 181 - 190.

Wetherill, G. B. (1963), "Sequential Estimation of Quantal Response Curves," *Journal of the Royal Statistical Society, Series B*, 25, 1 - 48.

Wu, C. F. J. (1985), "Efficient Sequential Designs With Binary Data," *Journal of the American Statistical Society*, 80, 974 - 983.

Wei, C. Z. (1985), "Asymptotic Properties of Least-Squares Estimates in Stochastic Regression Models," *The Annals of Statistics*,13, 1498 - 1508.

APPENDIX A

PROOF OF THEOREM 1

Theorem 1. Let $x_{1,1}, y_{1,1}, \ldots, x_{n,k}, y_{n,k}$ be a sequence of design levels and binary responses from SAM (9), where $G(x|\theta_1,\theta_2)$ is the two parameter logit expectation. Assume that the MLE, $(\hat{\theta}_1^{(n)}, \hat{\theta}_2^{(n)})$, converges almost surely to a constant, $(\theta_1^* \ \theta_2^*)$, $\theta_2^* \neq 0$. Also, assume that $M(x)$ is an increasing function of x. Then $\hat{L}_{p_1}^{(n)}$ and $\hat{L}_{p_2}^{(n)}$ from SAM (9) converge almost surely to L_{p_1} and L_{p_2}, respectively.

Proof: Since $(\hat{\theta}_1, \hat{\theta}_2)$ converges almost surely to (θ_1^*, θ_2^*), $x_{n+1,1}$ and $x_{n+1,2}$ converge almost surely to constants x_1^* and x_2^*, respectively. It will be demonstrated that $x_1^* = L_{p_1}$ and $x_2^* = L_{p_2}$.

From SAM (9), $G(x_{n+1,i}|\hat{\theta}_1^{(n)}, \hat{\theta}_2^{(n)}) = p_i$, $i = 1,2$. As n ---> ∞, this implies

$$G(x_1^*|\theta_1^*, \theta_2^*) = p_1 \quad \text{and} \quad G(x_2^*|\theta_1^*, \theta_2^*) = p_2. \tag{61}$$

From (19), the normal equations are

$$(1/n)\cdot\sum_{i}^{n}\sum_{j}^{2} G(x_{ij}|\theta_1,\theta_2) \quad = \quad (1/n)\cdot\sum_{i}^{n}\sum_{j}^{2} y_{ij}$$

$$(1/n)\cdot \sum_{i}^{n}\sum_{j}^{2} x_{ij}\cdot G(x_{ij}|\theta_1,\theta_2) \quad = \quad (1/n)\cdot\sum_{i}^{n}\sum_{j}^{2} x_{ij}\cdot y_{ij} \tag{62}$$

From (61), the left hand sides of the normal equations converge almost surely to $(p_1 + p_2)$ and $(x_1^* p_1 + x_2^* p_2)$, respectively. From the convergence of the MLEs and Theorem 1 of Dubins and Freedman (1965),

$$\frac{1}{n}\cdot\sum_{i}^{n}\sum_{j}^{2} y_{ij} \dashrightarrow M(x_1^*) + M(x_2^*) \text{ almost surely.} \tag{63}$$

Thus, the likelihood equations imply

$$p_1 + p_2 \quad = \quad M(x_1^*) + M(x_2^*) \quad \text{and}$$

$$x_1^*\cdot p_1 + x_2^*\cdot p_2 \quad = \quad x_1^*\cdot M(x_1^*) + x_2^*\cdot M(x_2^*). \tag{64}$$

Solving these equations yields $x_1^* = L_{p_1}$ and $x_2^* = L_{p_2}$.

Q.E.D.

APPENDIX B

PROOF OF THEOREM 2

Theorem 2. Let Y be a binary random variable with expectation M(x). Assume that $M(x) = G(x|\theta)$, where θ is a single unknown parameter. That is, the model, $G(x|\theta)$, used in SAM is the true expectation of Y. The design levels, x_i, are assumed to be such that $0 < K_1 < p_i = M(x_i) < K_2 < 1$, for some constants K_1, K_2. Assume the standard regularity conditions on the distribution of Y (given below) hold. If the following conditions are also satisfied, then $\hat{L}_p$ from SAM converges to L_p in probability.

1) $M(x|\theta)$ is continuous in θ.
2) $\exists\ \delta_1, \delta_2 \in R$, such that $\delta_1 < \partial M(x,\theta) / \partial\theta < \delta_2$, $\forall\ \theta$ in some neighborhood of the true value of θ, θ_0, and $\forall\ x \in (K_1, K_2)$
3) $\partial^2 M(x,\theta) / \partial\theta^2$ is bounded $\forall\ \theta$ in some neighborhood of the true value of θ, θ_0, and $\forall\ x \in (K_1, K_2)$

<u>Regularity Conditions</u>: from Serfling (1981),

Let Θ be an open interval in $\mathbb{R}$ and f(y) be the p.m.f. of Y.

a) For each $\theta \in \Theta$, the derivatives

$$\frac{\partial \log f(y;\theta)}{\partial\theta}, \quad \frac{\partial^2 \log f(y;\theta)}{\partial\theta^2}, \quad \frac{\partial^3 \log f(y;\theta)}{\partial\theta^3}$$

exist, all y;

b) For each $\theta_0 \in \Theta$, there exists functions g(y), h(y) and H(y) such that for θ in a neighborhood $N(\theta_0)$ the relations

$$\left|\frac{\partial \log f(y;\theta)}{\partial\theta}\right| \le g(y), \quad \left|\frac{\partial^2 \log f(y;\theta)}{\partial\theta^2}\right| \le h(y), \text{ and}$$

$$\left|\frac{\partial^3 \log f(y;\theta)}{\partial\theta^3}\right| \le H(y)$$

hold, all y, and

$\sum g(y) < \infty$, $\sum h(y) < \infty$, $E\{H(y)\} < \infty$ for $\theta \in N(\theta_0)$;

c) For each $\theta \in \Theta$,

$$0 < E\left[\left(\frac{\partial \log f(y,\theta)}{\partial\theta}\right)^2\right] < \infty.$$

proof:

Let $x_1,\ldots,x_n$ and $y_1,\ldots,y_n$ be the design levels and responses from SAM (9). By Section 5 of Wu (1985), $\hat{L}_p$ converges to L_p in probability, if the MLE, $\hat{\theta}$, converges to θ in probability. To prove $\hat{\theta}$ converges to θ_0, the results of Crowder (1975) are used. Following Crowder's notation,

$$\begin{aligned} l_n(\theta) &= \sum_1^n \ln\{P(y_i=1|x_1,y_1\ldots,y_{i-1})\} \\ &= \sum_1^n \left(y_i\cdot\ln\{p_i\} + (1-y_i)\cdot\ln\{1-p_i\} \right), \end{aligned} \tag{65}$$

$$l_n'(\theta) = \partial l_n(\theta)/\partial\theta = \sum_1^n (\partial p_i/\partial\theta)\cdot\frac{y_i - p_i}{p_i(1 - p_i)}, \tag{66}$$

$$l_n''(\theta) = \sum_1^n \left(\frac{y_i - p_i}{p_i(1-p_i)} \cdot \left[(\partial^2 p_i/\partial\theta^2)\cdot p_i \cdot (1-p_i) - \right. \right.$$

$$\left. \left. (\partial p_i/\partial\theta)^2 \cdot (1-2\cdot p_i) \right] - \frac{(\partial p_i/\partial\theta)^2}{p_i(1-p_i)} \right), \tag{67}$$

$$B_n = E\left(-l''(\theta)\right) = \sum_1^n E\left(\frac{(\partial p_i/\partial\theta)^2}{p_i(1-p_i)} \right). \tag{68}$$

By equation (2.3) of Crowder (1975), $\hat{\theta}_n$ converges in probability to θ, if $\exists\ \Delta > 0$ and a sequence $\{v_n\}$ tending to infinity, such that (69)

$$P\{-v_n^{-1/2}\cdot(\theta^* - \theta_0)^2 \cdot B_n^{1/2} \cdot l''(\theta^*) \geq \delta^2\} \text{ -----> } 1 \quad \text{as } n \text{ ---> } \infty,$$

when $|\theta^* - \theta_0| = \delta \leq \Delta$.

To show (69) holds, a lower bound for the expression $-B_n^{-1/2}\cdot l''(\theta^*)$ will be derived. Define $p_i^* = M(x_i, \theta^*)$. By the continuity of $M(x,\theta)$, for all $\varepsilon > 0$, $\exists\ \Delta$ such that $|p_i - p_i^*| < \varepsilon$. Define $\delta_{1'}$ $(\delta_{1'} < \delta_1)$ to be the lower bound of $(\partial p_i^* / \partial\theta) = \left(\partial M(x_i\theta) / \partial\theta\right)|_{\theta=\theta^*}$, for $\theta^* \in (\theta_0 - \Delta,\ \theta_0 + \Delta)$.

Now, $-l''(\theta^*)\cdot B_n^{-1/2} = \dfrac{T_1 - T_2}{T_3}$, where

$$T_1 = \sum_1^n \frac{(\partial p_i^*/\partial\theta)^2}{p_i^*\cdot(1-p_i^*)}, \tag{70}$$

$$T_2 = \sum_1^n \frac{y_i - p_i^*}{p_i^*\cdot(1-p_i^*)} \cdot \left[(\partial^2 p_i^*/\partial\theta^2)\cdot p_i^*(1-p_i^*) - (\partial p_i^*/\partial\theta)^2(1-2p_i^*) \right],$$

$$T_3 = \left[\sum_1^n E\left(\frac{(\partial p_i/\partial\theta)^2}{p_i\cdot(1-p_i)} \right) \right]^{1/2}.$$

Since $(\partial p_i^* / \partial \theta) > \delta_{1'}$,

$$T_1 \geq 4 \cdot n \cdot \delta_{1'}^2. \tag{71}$$

T_2 is equivalent to $\sum_1^n (y_i - p_i) \cdot q_i^* + \sum_1^n (p_i - p_i^*) \cdot q_i^*$, where

$$q_i^* = \frac{1}{p_i^{*2}(1-p_i^*)^2} \cdot \left[(\partial^2 p_i^* / \partial \theta^2) \cdot p_i^* (1-p_i^*) - (\partial p_i^* / \partial \theta)^2 (1-2 \cdot p_i^*) \right]. \tag{72}$$

By Theorem C of Serfling (1980,p. 27), $(1/n) \cdot \sum_1^n (y_i - p_i) \cdot q_i^*$ ----> 0, in probability. Also, Δ can be chosen small enough such that $(p_i - p_i^*) \cdot q_i^* < \delta_{1'}^2$, $\forall$ i. Therefore,

$$T_2 \leq n \cdot \delta_{1'}^2. \tag{73}$$

The random variable $\left(\frac{(\partial p_i / \partial \theta)^2}{p_i \cdot (1-p_i)} \right)$ is bounded, therefore, its expectation is bounded by a positive upper bound, say δ_3. Thus,

$$T_3 \leq (n \cdot \delta_3)^{1/2} \tag{74}$$

Using (70), (71), (72), and (74), as n ----> ∞,

$$-1''(\theta^*) \cdot B_n^{-1/2} \geq \frac{n \cdot (4 \cdot \delta_{1'}^2 - \delta_{1'}^2)}{(n \cdot \delta_3)^{1/2}} = \sqrt{n} \left(\frac{3 \cdot \delta_{1'}^2}{\delta_3^{1/2}} \right) \text{ ----> } \infty .$$

A sequence $\{v_n\}$ tending to infinity can now be chosen to satisfy (69). For example, let $\{v_n\} = \sqrt{n} \cdot \left(\frac{3 \cdot \delta_{1'}^2}{2 \cdot \delta_3^{1/2}} \right)$.

Therefore, $\hat{\theta}_n$ ---> θ_0 and $\hat{L}_p$ ----> L_p , in probability.

Q.E.D.

APPENDIX C

PROOF OF THEOREM 4

Theorem 3. Assuming $(x_{n,1}, x_{n,2})$ from SAM (9) converge almost surely to (L_p, L_{1-p}), $\hat{\lambda}_n$ as defined in (24) converges almost surely to $c_L = 2p(1-p)\cdot\ln[(1-p)/p] \;/\; (L_{1-p} - L_p)$.

Proof:

Unless otherwise specified, the summations in this proof on i run from 1,...,n, and for j run from 1 to 2. Rewrite (24) as

$$\hat{\lambda}_n = \lambda_n^* - \frac{c^*\cdot\sum(x_{i2} - x_{i1})}{\sum\sum(x_{ij} - \bar{x}_n)^2}, \tag{75}$$

where $\lambda_n^* = \dfrac{\sum\sum y_{ij}(x_{ij} - \bar{x}_n)}{\sum\sum(x_{ij} - \bar{x}_n)^2}$, $\bar{x}_n = \sum\sum x_{ij} \;/\; 2\cdot n$, and $c^* =$ $(1/2) - p - p(1-p)\cdot\ln\{(1-p)/p\}$. Let $\alpha = (1-2p)/(L_{1-p} - L_p)$ denote the slope of the line through (L_p, p) and $(L_{1-p}, 1-p)$. Assuming $x_{n,1} \xrightarrow{a.s.} L_p$ and $x_{n,2} \xrightarrow{a.s.} L_{1-p}$, the second term on the right hand side of (75) converges to $2\cdot c^*\cdot\alpha \;/\; (1-2p)$, since

$$\begin{aligned}\frac{c^*\cdot\sum(x_{i2} - x_{i1})}{\sum\sum(x_{ij} - \bar{x}_n)^2} \xrightarrow{a.s.} & \frac{c^* n\cdot(L_{1-p} - L_p)}{n(L_{1-p} - L_p)^2 \;/\; 2} \\ &= 2\cdot c^*\cdot\alpha \;/\; (1-2p).\end{aligned} \tag{76}$$

By Theorem 1 of Lai and Wei (1982) and Theorem 2 of Wei (1985), the first term of the right hand side side of (75), λ_n^*, converges almost surely to α if the eigenvalues, ξ, of

$$\begin{pmatrix} 2\cdot n & 0 \\ 0 & \sum\sum (x_{i,j} - \bar{x}_n)^2 \end{pmatrix} \tag{77}$$

satisfy

1. $\xi_{min}(n) \dashrightarrow \infty$ a.s.,
2. $\log \xi_{max}(n) = o(\xi_{min}(n))$ a.s.

However, conditions 1 and 2 are satisfied since

$$\sum_i\sum_j (x_{ij} - \bar{x}_n)^2 \tag{78}$$

$$= \sum_i (x_{i1} - \bar{x}_{n1})^2 + \sum_i (x_{i2} - \bar{x}_{n2})^2 + n(\bar{x}_{n1} - \bar{x}_n)^2 + n(\bar{x}_{n2} - \bar{x}_n)^2$$

$$= O(n) ,$$

where $\bar{x}_{nj} = \sum_i^n x_{ij} / n$, j=1,2. Therefore,

$$\lambda_n^* \xrightarrow{a.s.} \alpha \tag{79}$$

Since

$$\alpha\{1 - [2c^*/ (1-2p)\} = \frac{2p(1-p)\cdot \ln[(1-p)/p]}{L_{1-p} - L_p}, \tag{80}$$

(76) and (79) imply $\hat{\lambda}_n \xrightarrow{a.s.} c_L$. Q.E.D.

APPENDIX D

SAS CODE FOR THE SIMULATION STUDY

The following is the SAS code (SAS Institute, Inc.) used in the simulation study for initial procedure 1. The program actually produced the MSEs for each procedure in the n = 20 column of Table 9. The other values were generated by changing the sample size n and the formulas used to generate the binary responses.

```
DATA ONE;
**********************************************************;
**                  INITIALIZE VARIABLES               **;
**           N = # updates, M = # iterations           **;
**********************************************************;
  N =20; M = 500; MODEL = 'LOGIT'; SETUP = 2;
  DISCARD1 = 0; DISCARD2 = 0; DISCARD3 = 0;
  FILE PRINT; PUT ' SETUP ' SETUP N M MODEL;
  MSEW1_1 = 0; MSEW1_2 = 0; MSEW1_3 = 0;
  MSESM_1 = 0; MSESM_2; MSESM_3 = 0;
  MSEW2_1 = 0; MSEW2_2 = 0; MSEW2_3 = 0;
  MSEA1_1 = 0; MSEA1_2 = 0; MSEA1_3 = 0;
  MSEA2_1 = 0; MSEA2_2 = 0; MSEA2_3 = 0;
  MSERM_1 = 0; MSERM_2 = 0; MSERM_3 = 0;
  MSW1_1 = 0; MSW1_2 = 0; MSW1_3 = 0;
  MSSM_1 = 0; MSSM_2 = 0; MSSM_3 = 0;
  MSW2_1 = 0; MSW2_2 = 0; MSW2_3 = 0;
  MSA1_1 = 0; MSA1_2 = 0; MSA1_3 = 0;
  MSA2_1 = 0; MSA2_2 = 0; MSA2_3 = 0;
  MSRM_1 = 0; MSRM_2 = 0; MSRM_3 = 0;
  W1ADET_1 = 0; W1ADET_2 = 0; W2DET_1 = 0; W2DET_2 = 0;
  W1ADET_3 = 0; W1ADET_4 = 0; W2DET_3 = 0;
  W1BDET_1 = 0; W1BDET_2 = 0; W1BDET_3 = 0;
  SMDET_1 = 0; SMDET_2 = 0; SMDET_3 = 0;
  ARRAY X {42} X1-X42; ARRAY U {40} U1-U40;
  ARRAY Y {40} Y1-Y40;
```

```
*************************************************************;
***            START LOOP FOR 1000 SAMPLES               ***;
*************************************************************;
  DO L = 1 TO M;
     DO I = 1 TO (2*N);
       U{I} = RANUNI(35671);
     END;
*************************************************************;
***            INITIAL 10 POINTS (LOGIT MODEL)           ***;
*************************************************************;
    IF SETUP=1 THEN DO;
      X{1} = -2.1972246; X{2} = -.8472978; X{3} = -.8472978;
      X{4} = 0; X{5} = 0; X{6} = 0; X{7} = 0;
      X{8} = .8472978; X{9} = .8472978; X{10} = 2.1972246;
    END;
    IF SETUP=2 THEN DO;
      X{1} = -.8472978; X{2} = -.1603426; X{3} = -.1603426;
      X{4} = .241162; X{5} = .241162; X{6} = .241162;
      X{7} = .241162; X{8} = .5007752; X{9} = .5007752;
      X{10} = 1.3862944;
    END;
    ALPHA = .25; BETA = .5; CHK = 0;
    SUMX = 0; SUMY = 0; SUMXX = 0; SUMXY = 0;
*************************************************************;
***               LS SLOPE CALCULATIONS                   ***;
***               GENERATE 10 Y VALUES,                   ***;
***         AND CHECK SILVAPULLES CONDITIONS              ***;
*************************************************************;
    X0MIN = 50; X1MIN = 50;
    X0MAX = -50; X1MAX = -50;
    DO I = 1 TO 10;
      PROB = 1 / (1 + EXP(-X{I}));
      IF U{I} LT PROB THEN Y{I} = 1; ELSE Y{I} = 0;
      SUMX = SUMX + X{I}; SUMXX = SUMXX + X{I}*X{I};
      SUMY = SUMY + Y{I}; SUMXY = SUMXY + X{I}*Y{I};
      IF Y{I} = 0 THEN DO;
        IF X{I} GT X0MAX THEN X0MAX = X{I};
        IF X{I} LT X0MIN THEN X0MIN = X{I}; END;
      IF Y{I} = 1 THEN DO;
        IF X{I} GT X1MAX THEN X1MAX = X{I};
        IF X{I} LT X1MIN THEN X1MIN = X{I}; END;
    END;
***                  CONDITION 1                          ***;
    IF X1MAX GT X1MIN AND X0MIN LT X0MAX AND X1MAX GT X0MIN
    AND X1MIN LT X0MAX THEN CHK = 1;
***                  CONDITION 2                          ***;
    IF X0MIN = X0MAX AND X1MIN LT X0MIN AND X0MAX LT X1MAX
    THEN CHK = 1;
***                  CONDITION 3                          ***;
    IF X1MIN = X1MAX AND X0MIN LT X1MIN AND X1MAX LT X0MAX
    THEN CHK = 1;
    IF CHK = 0 THEN DO;
      DISCARD1 = DISCARD1 + 1; GO TO OK;
    END;
```

```
************************************************************;
***                   CALCULATE MLE'S                    ***;
************************************************************;
    G1=0; G2=0;
    DO J = 1 TO 10;
      IF (G1**2 + G2**2) LT .0001 AND J NE 1 THEN GO TO OK1;
      H11 = 0; H12 = 0; H22 = 0;
      G1 = 0; G2 = 0;
      DO I = 1 TO 10;
        Z = ALPHA + BETA*X{I};
        IF ABS(Z) GT 15 THEN DO;
          PRED=1; GO TO DE3;
        END;
        PRED = EXP(Z) / (1 + EXP(Z));
DE3:    H11 = H11 - PRED*(1-PRED);
        H12 = H12 - X{I}*PRED*(1-PRED);
        H22 = H22 - X{I}*X{I}*PRED*(1-PRED);
        G1 = G1 + (Y{I} - PRED);
        G2 = G2 + (X{I}*Y{I} - X{I}*PRED);
      END;
      IF (H11*H22) - (H12**2) LT .001 THEN DO;
        DISCARD3 = DISCARD3 + 1; GO TO OK;
      END;
      DET = (H11*H22) - (H12**2);
      HINV11 = H22 / DET;
      HINV22 = H11 / DET;
      HINV12 = -(H12 / DET);
      ALPHA = ALPHA - ((HINV11*G1) + (HINV12*G2));
      BETA  = BETA  - ((HINV12*G1) + (HINV22*G2));
    END;
***********************************************************;
***            CALCULATE STARTING VALUES                ***;
***********************************************************;
OK1: IF BETA LE 0 THEN DO;
      DISCARD2 = DISCARD2 + 1; GO TO OK;
    END;
    XL = (LOG(.25) - ALPHA) / BETA;
    XM = -ALPHA / BETA;
    XU = (LOG(4) - ALPHA) / BETA;
    IF XL LT -5 THEN XL = -5; IF XU GT 5 THEN XU = 5;
    IF ABS(XM) GT 5 THEN XM = SIGN(XM)*5;
    P20 = 1 / (1 + EXP(-XL)); P50 = 1 / (1 + EXP(-XM));
    P80 = 1 / (1 + EXP(-XU)); ALPH = ALPHA;
    BET = BETA; ALPHAN= -ALPHA/BETA; BETAN=BETA;
    XBAR = SUMX / 10;
    LSBETA = (SUMXY - XBAR*SUMY) / (SUMXX - (SUMX**2)/10);
    C = LOG(3) / LOG(.25);
*;
*;
*;
***********************************************************;
***      ROBBINS_MONRO PROCEDURE   (2 INDEP PROCS)     ****;
***********************************************************;
***           RUN FOR EACH RM CONSTANT C                ***;
```

```
DO RMC = 1, 6, 36;
  X{11} = XL;
  P20 = 1 / (1 + EXP(-XL));
************************************************************;
***            1RST PROCESS (RUN AT P = 20)            ***;
************************************************************;
  DO I = 11 TO (N+5);
***                GENERATE THE Y'S                    ***;
    IF U{I} LT P20 THEN Y{I} = 1; ELSE Y{I} = 0;
***                CALCULATE NEXT X'S                  ***;
    X{I + 1} = X{I} - RMC*(Y{I} - .2) / I;
    IF X{I + 1} GT 5 THEN X{I + 1} = 5;
    IF X{I + 1} LT -5 THEN X{I + 1} = -5;
    P20 = 1 / (1 + EXP(-X{I + 1}));
  END;
***          GENERATE FINAL ESTIMATE FOR P = 20        ***;
  XRM20F = X{N+6};
*;
************************************************************;
***             2ND PROCESS (RUN AT P = 80)            ***;
************************************************************;
  X{11} = XU;
  P80 = 1 / (1 + EXP(-XU));
  DO I = 11 TO (N+5);
***                GENERATE THE Y'S                    ***;
    IF U{I} LT P80 THEN Y{I} = 1; ELSE Y{I} = 0;
***                 GENERATE NEXT X'S                  ***;
    X{I + 1} = X{I} - RMC*(Y{I} - .8) / I;
    IF X{I + 1} GT 5 THEN X{I + 1} = 5;
    IF X{I + 1} LT -5 THEN X{I + 1} = -5;
    P80 = 1 / (1 + EXP(-X{I + 1}));
  END;
***         GENERATE FINAL ESTIMATE FOR P = 80         ***;
  XRM80F = X{N+6};
***;
************************************************************;
***              GENERATE FINAL ESTIMATE               ***;
***              FROM 2 INDEP ESTIMATES, AND           ***;
***              UPDATE THE MEAN SQUARE ERROR          ***;
************************************************************;
  XRM50F = (.5)*(XRM20F + XRM80F);
  XRM75F = (.5)*((1 - C)*XRM80F + (1 + C)*XRM20F);
  IF RMC = 1 THEN DO;  MSERM_1 = MSERM_1 + XRM50F**2;
    MSRM_1 = MSRM_1 + (XRM75F - LOG(3))**2;
  END;
  IF RMC = 6 THEN DO;  MSERM_2 = MSERM_2 + XRM50F**2;
     MSRM_2 = MSRM_2 + (XRM75F - LOG(3))**2;
  END;
  IF RMC = 36 THEN DO;  MSERM_3 = MSERM_3 + XRM50F**2;
    MSRM_3 = MSRM_3 + (XRM75F - LOG(3))**2;
  END;
END;
*;
*;
```

```
*;
**********************************************************;
***        ANBAR'S PROCEDURE   (2 INDEP PROCS)        ***;
**********************************************************;
***              RUN FOR EACH UPPER BOUND             ***;
DO UB = 10, 50, 100;
  X{11} = XL;
  P20 = 1 / (1 + EXP(-XL)); P80 = 1 / (1 + EXP(-XU));
  SUMXYA = SUMXY - (.2)*SUMX; SUMYA = SUMY - 2;
  SUMXXA = SUMXX; SUMXA = SUMX;
**********************************************************;
***            1RST POINT (RUN AT P = 20)             ***;
**********************************************************;
  DO I = 11 TO (N+5);
***                GENERATE THE Y'S                   ***;
    IF U{I} LT P20 THEN Y{I} = 1; ELSE Y{I} = 0;
***          CALCULATE THE NEW LS SLOPE               ***;
    SUMXA = SUMXA + X{I};
    SUMYA = SUMYA + (Y{I}-.2);
    SUMXYA = SUMXYA + X{I}*(Y{I}-.2);
    SUMXXA = SUMXXA + X{I}*X{I};
    XBARA = SUMXA / I;
    BETAAN = (SUMXYA-XBARA*SUMYA) / (SUMXXA-(SUMXA**2)/I);
***            CHECK THE BOUNDS ON LS SLOPE           ***;
    IF BETAAN NE 0 THEN DSTAR = 1 / BETAAN;
    IF BETAAN = 0 OR (1/BETAAN) GT UB THEN DSTAR=UB;
    IF BETAAN LT 0 THEN DSTAR=UB;
***                GENERATE NEXT X'S                  ***;
    X{I + 1} = X{I} - DSTAR*(Y{I} - .2)/I;
    IF X{I + 1} GT 5 THEN X{I + 1} = 5;
    IF X{I + 1} LT -5 THEN X{I + 1} = -5;
    P20 = 1 / (1 + EXP(-X{I + 1}));
  END;
***            GENERATE FINAL ESTIMATE FOR P = 20     ***;
  XAN20F = X{N+6};
*;
**********************************************************;
***             2ND PROC (RUN AT P = 80)              ***;
**********************************************************;
  X{11} = XU;
  P20 = 1 / (1 + EXP(-XL)); P80 = 1 / (1 + EXP(-XU));
  SUMXYA = SUMXY - (.8)*SUMX; SUMYA = SUMY - 8;
  SUMXA=SUMX; SUMXXA = SUMXX;
  DO I = 11 TO (N+5);
***                GENERATE THE Y'S                   ***;
     IF U{I} LT P80 THEN Y{I} = 1; ELSE Y{I} = 0;
***          CALCULATE THE NEW LS SLOPE               ***;
    SUMXA = SUMXA + X{I};
    SUMYA = SUMYA + (Y{I}-.8);
    SUMXYA = SUMXYA + X{I}*(Y{I}-.8);
    SUMXXA = SUMXXA + X{I}*X{I};
    XBARA = SUMXA / I;
    BETAAN = (SUMXYA-XBARA*SUMYA) / (SUMXXA-(SUMXA**2)/I);
***            CHECK THE BOUNDS ON LS SLOPE           ***;
```

```
    IF BETAAN NE 0 THEN DSTAR = 1 / BETAAN;
    IF BETAAN = 0 OR (1/BETAAN) GT UB THEN DSTAR=UB;
    IF BETAAN LT 0 THEN DSTAR=UB;
***                GENERATE NEXT X'S                      ***;
    X{I + 1} = X{I} - DSTAR*(Y{I} - .8)/I;
    IF X{I + 1} GT 5 THEN X{I + 1} = 5;
    IF X{I + 1} LT -5 THEN X{I + 1} = -5;
    P80 = 1 / (1 + EXP(-X{I + 1}));
  END;
 ***          GENERATE FINAL ESTIMATE FOR 80              ***;
  XAN80F = X{N+6};
 ***;
 ***********************************************************;
 ***              GENERATE FINAL ESTIMATE                 ***;
 ***              FROM 2 INDEP ESTIMATES, AND             ***;
 ***              UPDATE THE MEAN SQUARE ERROR            ***;
 ***********************************************************;
  XAN50F = (.5)*(XAN20F + XAN80F);
  XAN75F = (.5)*((1 - C)*XAN80F + (1 + C)*XAN20F);
  IF UB = 10 THEN DO;  MSEA1_1 = MSEA1_1 + XAN50F**2;
    MSA1_1 = MSA1_1 + (XAN75F - LOG(3))**2;
  END;
  IF UB = 50 THEN DO;  MSEA1_2 = MSEA1_2 + XAN50F**2;
    MSA1_2 = MSA1_2 + (XAN75F - LOG(3))**2;
  END;
  IF UB = 100 THEN DO;  MSEA1_3 = MSEA1_3 + XAN50F**2;
    MSA1_3 = MSA1_3 + (XAN75F - LOG(3))**2;
  END;
 END;
 *;
 *;
 *;
 ***********************************************************;
 ***         WU'S PROCEDURE   (2 INDEP PROCS)             ***;
 ***********************************************************;
 ***               RUN FOR EACH UPPER BOUND               ***;
 DO UB = 10, 50, 100;
  ALPHA = ALPH; BETA = BET; X{11} = XL;
  P20 = 1 / (1 + EXP(-XL)); P50 = 1 / (1 + EXP(-XM));
  P80 = 1 / (1 + EXP(-XU));
 ***********************************************************;
 ***            1RST POINT (RUN AT P = 20)                ***;
 ***********************************************************;
  DO I = 11 TO (N+5);
 ***               GENERATE THE Y'S                       ***;
    IF U{I} LT P20 THEN Y{I} = 1; ELSE Y{I} = 0;
 ***             CALCULATE THE NEW MLE'S                  ***;
    G1=0; G2=0;
    DO J=1 TO 10;
      IF (G1**2 + G2**2) LT .0001 AND J NE 1 THEN GO TO OK2;
      H11 = 0; H12 = 0; H22 = 0;
      G1 = 0; G2 = 0;
      DO K=1 TO I;
        Z = ALPHA + BETA*X{K};
```

```
        IF ABS(Z) GT 15 THEN DO;
          PRED=1; GO TO CK1;
        END;
        PRED = EXP(Z) / (1 + EXP(Z));
CK1:    H11 = H11 - PRED*(1-PRED);
        H12 = H12 - X{K}*PRED*(1-PRED);
        H22 = H22 - X{K}*X{K}*PRED*(1-PRED);
        G1 = G1 + (Y{K} - PRED);
        G2 = G2 + (X{K}*Y{K} - X{K}*PRED);
      END;
      IF ABS((H11*H22) - (H12**2)) LT .001 THEN DO;
        DSTAR=0; GO TO CKL1;
      END;
      DET = (H11*H22) - (H12**2);
      HINV11 = H22 / DET;
      HINV22 = H11 / DET;
      HINV12 = -(H12 / DET);
      ALPHA = ALPHA - ((HINV11*G1) + (HINV12*G2));
      BETA  = BETA  - ((HINV12*G1) + (HINV22*G2));
    END;
***       CHECK THE BOUNDS ON BETAWU (SLOPE)              ***;
OK2:  X{I+1} = (LOG(.25) - ALPHA)/BETA;
    DSTAR = (X{I} - X{I+1})*I / (Y{I} - .2);
    IF DSTAR GT UB THEN DSTAR=UB;
    IF DSTAR LT (-UB) THEN DSTAR=(-UB);
    IF BETA LT 0 THEN DSTAR=UB;
***               GENERATE NEXT X'S                       ***;
CKL1:  X{I + 1} = X{I} - (DSTAR/I)*(Y{I} - .2);
    IF X{I+1} GT 5 THEN X{I+1}=5;
    IF X{I+1} LT -5 THEN X{I+1}=-5;
    P20 = 1 / (1 + EXP(-X{I + 1}));
  END;
***        GENERATE FINAL ESTIMATE FOR P = 20             ***;
  XWU20F = X{N+6};
*;
**********************************************************;
***             2ND POINT (RUN FOR P = 80)                ***;
**********************************************************;
  X{11} = XU; ALPHA = ALPH; BETA = BET;
  DO I = 11 TO (N+5);
***               GENERATE THE Y'S                        ***;
    IF U{I} LT P80 THEN Y{I} = 1; ELSE Y{I} = 0;
***             CALCULATE THE NEW MLE'S                   ***;
    G1=0; G2=0;
    DO J=1 TO 10;
      IF (G1**2 + G2**2) LT .0001 AND J NE 1 THEN GO TO OK3;
      H11 = 0; H12 = 0; H22 = 0;
      G1 = 0; G2 = 0;
      DO K=1 TO I;
        Z = ALPHA + BETA*X{K};
        IF ABS(Z) GT 15 THEN DO;
          PRED=1; GO TO CK2;
        END;
        PRED = EXP(Z) / (1 + EXP(Z));
```

```
CK2:     H11 = H11 - PRED*(1-PRED);
         H12 = H12 - X{K}*PRED*(1-PRED);
         H22 = H22 - X{K}*X{K}*PRED*(1-PRED);
         G1 = G1 + (Y{K} - PRED);
         G2 = G2 + (X{K}*Y{K} - X{K}*PRED);
       END;
       IF ABS((H11*H22) - (H12**2)) LT .001 THEN DO;
         DSTAR=0; GO TO CKL2;
       END;
      DET = (H11*H22) - (H12**2);
       HINV11 = H22 / DET;
       HINV22 = H11 / DET;
       HINV12 = -(H12 / DET);
       ALPHA = ALPHA - ((HINV11*G1) + (HINV12*G2));
       BETA  = BETA  - ((HINV12*G1) + (HINV22*G2));
     END;
***         CHECK THE BOUNDS ON BETAWU (SLOPE)              ***;
OK3:  X{I+1} = (LOG(4) - ALPHA)/BETA;
     DSTAR = (X{I} - X{I+1})*I / (Y{I} - .8);
     IF DSTAR GT UB THEN DSTAR=UB; END;
     IF DSTAR LT (-UB) THEN DSTAR=(-UB);
     IF BETA LT 0 THEN DSTAR=UB;
***                 GENERATE NEXT X'S                       ***;
CKL2:  X{I + 1} = X{I} - (DSTAR/I)*(Y{I} - .8);
     IF X{I+1} GT 5 THEN X{I+1}=5;
     IF X{I+1} LT -5 THEN X{I+1}=-5;
     P80 = 1 / (1 + EXP(-X{I + 1}));
   END;
***            GENERATE FINAL ESTIMATE FOR 80               ***;
   XWU80F = X{N+6};
***;
*****************************************************************;
***                 GENERATE FINAL ESTIMATE                     ***;
***                 FROM 2 INDEP ESTIMATES, AND                 ***;
***                 UPDATE THE MEAN SQUARE ERROR                ***;
*****************************************************************;
   XWU1F50 = (.5)*(XWU20F + XWU80F);
   XWU1F75 = (.5)*((1 - C)*XWU80F + (1 + C)*XWU20F);
   IF UB = 10 THEN DO;  MSEW1_1 = MSEW1_1 + XWU1F50**2;
     MSW1_1 = MSW1_1 + (XWU1F75 - LOG(3))**2;
   END;
   IF UB = 50 THEN DO;  MSEW1_2 = MSEW1_2 + XWU1F50**2;
     MSW1_2 = MSW1_2 + (XWU1F75 - LOG(3))**2;
   END;
   IF UB = 100 THEN DO; MSEW1_3 = MSEW1_3 + XWU1F50**2;
     MSW1_3 = MSW1_3 + (XWU1F75 - LOG(3))**2;
   END;
END;
*;
*;
*;
***********************************************************;
***      ANBAR'S PROCEDURE   (2N POINTS AT P = .5)      ***;
***********************************************************;
```

```
*;
***              RUN FOR EACH UPPER BOUND                    ***;
DO UB = 10, 50, 100;
  X{11} = XM; P50 = 1 / (1 + EXP(-XM));
  SUMXYA = SUMXY; SUMXA = SUMX; SUMYA = SUMY;
  SUMXXA = SUMXX; XBARA=0;
  DO I = 11 TO (2*N);
***                GENERATE THE Y'S                          ***;
    IF U{I} LT P50 THEN Y{I} = 1; ELSE Y{I} = 0;
***           CALCULATE THE NEW LS SLOPE                     ***;
    SUMXA = SUMXA + X{I};
    SUMYA = SUMYA + (Y{I} - .5);
    SUMXYA = SUMXYA + X{I}*(Y{I} - .5);
    SUMXXA = SUMXXA + X{I}*X{I};
    XBARA = SUMXA / I;
    BETAAN = (SUMXYA-XBARA*SUMYA) / (SUMXXA-(SUMXA**2)/I);
***          CHECK THE BOUNDS ON BETAAN (LS SLOPE)          ***;
    IF BETAAN NE 0 THEN DSTAR = 1 / BETAAN;
    IF BETAAN = 0 OR DSTAR GT UB THEN DSTAR=UB;
    IF DSTAR LT 0 THEN DSTAR=0;
    IF BETAAN LT 0 THEN DSTAR=UB;
***                GENERATE NEXT X'S                        ***;
    X{I + 1} = X{I} - DSTAR*(Y{I} - .5)/I;
    IF X{I + 1} GT 5 THEN X{I + 1} = 5;
    IF X{I + 1} LT -5 THEN X{I + 1} = -5;
    P50 = 1 / (1 + EXP(-X{I + 1}));
  END;
***              GENERATE FINAL ESTIMATE                     ***;
  XAN50F = X{2*N + 1};
  IF BETAAN LE 0 OR (1/BETAAN) GT UB THEN BSTAR = 1/UB;
    ELSE BSTAR=BETAAN;
  XAN75F = X{2*N+1} + LOG(3)/(4*BSTAR);
  IF XAN75F GT 5 THEN  XAN75F = 5;
***             UPDATE MEAN SQUARE ERROR                     ***;
  IF UB = 10 THEN DO; MSEA2_1 = MSEA2_1 + XAN50F**2;
    MSA2_1 = MSA2_1 + (XAN75F - LOG(3))**2;
  END;
  IF UB = 50 THEN DO; MSEA2_2 = MSEA2_2 + XAN50F**2;
    MSA2_2 = MSA2_2 + (XAN75F - LOG(3))**2;
  END;
  IF UB = 100 THEN DO; MSEA2_3 = MSEA2_3 + XAN50F**2;
    MSA2_3 = MSA2_3 + (XAN75F - LOG(3))**2;
  END;
END;
*;
*;
*;
*************************************************************;
***                    SAM PROCEDURE                         ***;
*************************************************************;
***             GET STARTING VALUES AND                      ***;
***             RUN FOR EACH UPPER BOUND                     ***;
DO UB = 10, 50, 100;
  X{11} = XL; X{12} = XU;
```

```
  ALPHA = ALPH; BETA = BET;
  P20 = 1 / (1 + EXP(-XL)); P80 = 1 / (1 + EXP(-XU));
  DO I = 1 TO (N - 5);
***              GENERATE THE Y'S                        ***;
    IF U{10 + 2*I - 1} LT P20 THEN Y{10 + 2*I - 1} = 1;
      ELSE Y{10 + 2*I - 1} = 0;
    IF U{10 + 2*I} LT P80 THEN Y{10 + 2*I} = 1;
      ELSE Y{10 + 2*I} = 0;
***            CALCULATE THE NEW MLE'S                   ***;
    G1=0; G2=0;
    DO J=1 TO 10;
      IF (G1**2 + G2**2) LT .0001 AND J NE 1 THEN GO TO OK5;
      H11 = 0; H12 = 0; H22 = 0;
      G1 = 0; G2 = 0;
      DO K=1 TO (10 + 2*I);
        Z = ALPHA + BETA*X{K};
        IF ABS(Z) GT 15 THEN DO; PRED=1; GO TO CK3; END;
        PRED = EXP(Z) / (1 + EXP(Z));
CK3:    H11 = H11 - PRED*(1-PRED);
        H12 = H12 - X{K}*PRED*(1-PRED);
        H22 = H22 - X{K}*X{K}*PRED*(1-PRED);
        G1 = G1 + (Y{K} - PRED);
        G2 = G2 + (X{K}*Y{K} - X{K}*PRED);
      END;
      IF ABS((H11*H22) - (H12**2)) LT .001 THEN DO;
        DSTAR=0; DSTAR2=0; GO TO CKL3;
      END;
      DET = (H11*H22) - (H12**2);
      HINV11 = H22 / DET;
      HINV22 = H11 / DET;
      HINV12 = -(H12 / DET);
      ALPHA = ALPHA - ((HINV11*G1) + (HINV12*G2));
      BETA  = BETA  - ((HINV12*G1) + (HINV22*G2));
    END;
***         CHECK THE BOUNDS ON SLOPE (BETASM)           ***;
OK5:  X{10+2*(I+1)-1} = (LOG(.25) - ALPHA)/BETA;
    X{10+2*(I+1)} = (LOG(4) - ALPHA)/BETA;
    DSTAR = (X{10+2*I-1} -
            X{10+2*(I+1)-1})*(10+2*I-1)/(Y{10+2*I-1} - .2);
    DSTAR2 = (X{10+2*I} -
             X{10+2*(I+1)})*(10+2*I) / (Y{10+2*I} - .8);
    IF DSTAR GT UB THEN DSTAR=UB;
    IF DSTAR LT (-UB) THEN DSTAR=(-UB);
    IF DSTAR2 GT UB THEN DSTAR=UB;
    IF DSTAR2 LT (-UB) THEN DSTAR=(-UB);
    IF BETA LT 0 THEN DO; DSTAR=UB; DSTAR2=UB; END;
***               GENERATE NEXT X'S                      ***;
CKL3: X{10+2*(I+1)-1} = X{10+2*I-1} -
                      (DSTAR/(10+2*I-1))*(Y{10+2*I-1}-.2);
    X{10+2*(I+1)} = X{10+2*I} -
                     (DSTAR2/(10+2*I))*(Y{10+2*I}-.8);
    IF X{10+2*(I+1)-1} GT 5 THEN X{10+2*(I+1)-1} = 5;
    IF X{10+2*(I+1)} GT 5 THEN X{10+2*(I+1)} = 5;
    IF X{10+2*(I+1)-1} LT -5 THEN X{10+2*(I+1)-1} = -5;
```

```
    IF X{10+2*(I+1)} LT -5 THEN X{10+2*(I+1)} = -5;
    P20 = 1 / (1 + EXP(-X{10 + 2*(I + 1) - 1}));
    P80 = 1 / (1 + EXP(-X{10 + 2*(I + 1)}));
  END;
***              CALCULATE FINAL ESTIMATE                  ***;
  IF (1/BETA) GT UB OR BETA LE 0 THEN BSTAR = 1/UB;
    ELSE BSTAR = BETA;
  XSMF50=-ALPHA / BETA;
  XSMF75 = XSMF50 + (LOG(3)/BSTAR);
  IF XSMF75 GT 5 THEN  XSMF75 = 5;
***              UPDATE MEAN SQUARED ERROR                 ***;
  IF UB = 10 THEN DO; MSESM_1 = MSESM_1 + (XSMF50**2);
    MSSM_1 = MSSM_1 + (XSMF75 - LOG(3))**2;
  END;
  IF UB = 50 THEN DO; MSESM_2 = MSESM_2 + (XSMF50**2);
    MSSM_2 = MSSM_2 + (XSMF75 - LOG(3))**2;
  END;
  IF UB = 100 THEN DO; MSESM_3 = MSESM_3 + (XSMF50**2);
    MSSM_3 = MSSM_3 + (XSMF75 - LOG(3))**2;
  END;
END;
*;
*;
*;
***********************************************************;
***       WU'S PROCEDURE   (2N AT ONE POINT)               ***;
***********************************************************;
***             RUN FOR EACH UPPER BOUND                   ***;
DO UB = 10, 50, 100;
  ALPHA = ALPH; BETA = BET;
  X{11} = XM; P50 = 1 / (1 + EXP(-XM));
  DO I = 11 TO 2*N;
***                GENERATE THE Y'S                       ***;
    IF U{I} LT P50 THEN Y{I} = 1; ELSE Y{I} = 0;
***              CALCULATE THE NEW MLE'S                  ***;
    G1=0; G2=0;
    DO J=1 TO 10;
      IF (G1**2 + G2**2) LT .0001 AND J NE 1 THEN GO TO OK4;
      H11 = 0; H12 = 0; H22 = 0;
      G1 = 0; G2 = 0;
      DO K=1 TO I;
        Z = ALPHA + BETA*X{K};
IF ABS(Z) GT 15 THEN DO; PRED=1; GO TO CK4; END;
        PRED = EXP(Z) / (1 + EXP(Z));
CK4:    H11 = H11 - PRED*(1-PRED);
        H12 = H12 - X{K}*PRED*(1-PRED);
        H22 = H22 - X{K}*X{K}*PRED*(1-PRED);
        G1 = G1 + (Y{K} - PRED);
        G2 = G2 + (X{K}*Y{K} - X{K}*PRED);
      END;
      IF ABS((H11*H22) - (H12**2)) LT .001 THEN DO;
        DSTAR=0; GO TO CKL4;
      END;
      DET = (H11*H22) - (H12**2);
```

```
      HINV11 = H22 / DET;
      HINV22 = H11 / DET;
      HINV12 = -(H12 / DET);
      ALPHA = ALPHA - ((HINV11*G1) + (HINV12*G2));
      BETA  = BETA  - ((HINV12*G1) + (HINV22*G2));
    END;
 ***        CHECK THE BOUNDS ON SLOPE (BETAWU)            ***;
 OK4:  X{I+1} = -ALPHA/BETA;
     DSTAR = (X{I} - X{I+1})*I / (Y{I} - .5);
     IF DSTAR GT UB THEN DSTAR=UB;
     IF DSTAR LT (-UB) THEN DSTAR=(-UB);
     IF BETA LT 0 THEN DSTAR=UB;
 ***              GENERATE NEXT X'S                       ***;
CKL4: X{I + 1} = X{I} - (DSTAR/I)*(Y{I} - .5);
    IF X{I+1} GT 5 THEN X{I+1} = 5;
    IF X{I+1} LT -5 THEN X{I+1} = -5;
    P50 = 1 / (1 + EXP(-X{I + 1}));
   END;
 ***            GENERATE FINAL ESTIMATE                   ***;
   XWU2F50 = X{2*N+1};
   IF BETA LE 0 OR (1/BETA) GT UB THEN BSTAR=1/UB;
     ELSE BSTAR=BETA;
   XWU2F75 = X{2*N+1} + (LOG(3)/BSTAR);
   IF XWU2F75 GT 5 THEN XWU2F75=5;
 ***             UPDATE MEAN SQUARE ERROR                 ***;
   IF UB = 10 THEN DO; MSEW2_1 = MSEW2_1 + XWU2F50**2;
     MSW2_1 = MSW2_1 + (XWU2F75 - LOG(3))**2;
   END;
   IF UB = 50 THEN DO; MSEW2_2 = MSEW2_2 + XWU2F50**2;
     MSW2_2 = MSW2_2 + (XWU2F75 - LOG(3))**2;
   END;
   IF UB = 100 THEN DO; MSEW2_3 = MSEW2_3 + XWU2F50**2;
     MSW2_3 = MSW2_3 + (XWU2F75 - LOG(3))**2;
   END;
 END;
 OK:  END;
 **********************************************************;
 ***               CALCULATE FINAL MSEs                   ***;
 **********************************************************;
 PUT '---------------------------------------------------';
 PUT '---------------------------------------------------';
 PUT 'DISCARDS ARE ' DISCARD1 DISCARD2 DISCARD3;
 PUT '---------------------------------------------------';
 MSERM_1 = MSERM_1 / (M - DISCARD1 - DISCARD2 - DISCARD3);
 MSERM_2 = MSERM_2 / (M - DISCARD1 - DISCARD2 - DISCARD3);
 MSERM_3 = MSERM_3 / (M - DISCARD1 - DISCARD2 - DISCARD3);
 MSRM_1 = MSRM_1 / (M - DISCARD1 - DISCARD2 - DISCARD3);
 MSRM_2 = MSRM_2 / (M - DISCARD1 - DISCARD2 - DISCARD3);
 MSRM_3 = MSRM_3 / (M - DISCARD1 - DISCARD2 - DISCARD3);
 PUT 'MSES FOR RM P=50 ARE ' MSERM_1 MSERM_2 MSERM_3;
 PUT 'MSES FOR RM P=75 ARE ' MSRM_1 MSRM_2 MSRM_3;
 PUT '---------------------------------------------------';
 MSEA2_1 = MSEA2_1 / (M - DISCARD1 - DISCARD2 - DISCARD3);
 MSEA2_2 = MSEA2_2 / (M - DISCARD1 - DISCARD2 - DISCARD3);
```

```
MSEA2_3 = MSEA2_3 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSA2_1 = MSA2_1 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSA2_2 = MSA2_2 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSA2_3 = MSA2_3 / (M - DISCARD1 - DISCARD2 - DISCARD3);
PUT 'MSES FOR AN-2 P=50 ARE ' MSEA2_1 MSEA2_2 MSEA2_3;
PUT 'MSES FOR AN-2 P=75 ARE ' MSA2_1 MSA2_2 MSA2_3;
PUT '---------------------------------------------------';
MSEW1_1 = MSEW1_1 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSEW1_2 = MSEW1_2 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSEW1_3 = MSEW1_3 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSW1_1 = MSW1_1 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSW1_2 = MSW1_2 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSW1_3 = MSW1_3 / (M - DISCARD1 - DISCARD2 - DISCARD3);
PUT 'MSES FOR WU1 P=50 ARE ' MSEW1_1 MSEW1_2 MSEW1_3;
PUT 'MSES FOR WU1 P=75 ARE ' MSW1_1 MSW1_2 MSW1_3;
PUT '---------------------------------------------------';
MSEA1_1 = MSEA1_1 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSEA1_2 = MSEA1_2 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSEA1_3 = MSEA1_3 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSA1_1 = MSA1_1 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSA1_2 = MSA1_2 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSA1_3 = MSA1_3 / (M - DISCARD1 - DISCARD2 - DISCARD3);
PUT 'MSES FOR AN-1 P=50 ARE ' MSEA1_1 MSEA1_2 MSEA1_3;
PUT 'MSES FOR AN-1 P=75 ARE ' MSA1_1 MSA1_2 MSA1_3;
PUT '---------------------------------------------------';
MSESM_1 =MSESM_1 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSESM_2 =MSESM_2 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSESM_3 =MSESM_3 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSSM_1 =MSSM_1 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSSM_2 =MSSM_2 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSSM_3 =MSSM_3 / (M - DISCARD1 - DISCARD2 - DISCARD3);
PUT 'MSES FOR SM P=50 ARE ' MSESM_1 MSESM_2 MSESM_3;
PUT 'MSES FOR SM P=75 ARE ' MSSM_1 MSSM_2 MSSM_3;
PUT '---------------------------------------------------';
MSEW2_1 = MSEW2_1 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSEW2_2 = MSEW2_2 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSEW2_3 = MSEW2_3 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSW2_1 = MSW2_1 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSW2_2 = MSW2_2 / (M - DISCARD1 - DISCARD2 - DISCARD3);
MSW2_3 = MSW2_3 / (M - DISCARD1 - DISCARD2 - DISCARD3);
PUT 'MSES FOR WU-2 P=50 ARE ' MSEW2_1 MSEW2_2 MSEW2_3;
PUT 'MSES FOR WU-2 P=75 ARE ' MSW2_1 MSW2_2 MSW2_3;
PUT '---------------------------------------------------';
```

APPENDIX E

USER PROGRAMS

Four programs have been written to assist a researcher in using SAM. The two parameter logit model is used for $G(x|\theta)$. Also, (.2,.8) is chosen for (p_1,p_2). In this appendix, a brief description of each program along with the SAS codes are presented.

INITIAL

SAM's updating rule requires that MLEs exist. Thus, some procedure is needed to generate the initial design levels. The program INITIAL uses the Two Dimensional RM procedure of Moser and Fei (1989a) to calculate design levels until MLEs exist. The first step in this procedure is to observe the responses at the initial estimates of the 20^{th} and 80^{th} percentiles. New design levels are then calculated using the updating rules of the Two Dimensional RM procedure. Conditions for the existence of MLEs are checked at each update. If the MLEs exist for the data, a message is produced instructing the user to switch to SAM's updating rules (and to the program NEXTSAM).

MLECHECK

The program MLECHECK is designed for when a researcher already has an initial set of data, but has not used SAM. The program MLECHECK reads an askii data set and determines whether the MLEs of θ_1 and θ_2 exist. The user enters the name of the dataset (extension and filename). The dataset should contain three variables, the observation number, design level, and response, in column format. For example, with the data in the example of Chapter III,

1	2.0	0
2	4.0	0
3	2.0	0
4	4.5	1
5	3.0	0
6	4.75	0
7	3.0	1
8	5.0	1
9	4.0	0
10	5.0	1

If the MLEs exists, then a message is produced instructing the user to run the program NEXTSAM, to obtain the next design levels. If the MLEs do not exist, design levels to help obtain existence are suggested. A warning is given if the MLEs exist and the slope estimate, $\hat{\theta}_2$, is negative. When the program was run with the above data, a message "MLE exists, run NEXTSAM program" was produced on the screen.

NEXTSAM

The program, NEXTSAM, calculates the design levels using SAM's updating rules. Prior to using NEXTSAM for the first time, the user should verify that MLEs exist for the set of data (using INITIAL or MLECHECK). The user enters upper and lower bounds on the design levels, as well as initial estimates of the 20^{th} and 80^{th} percentiles. The upper and lower bounds should be chosen such that a response almost always occurs at the upper bound and almost never at the lower bound. A graph of the estimated expectation curve, $\hat{G}_n(x)$, is produced along with the design levels for the next update. Figure 3 on page 24 is an example of the graph produced by NEXTSAM.

SAMSUMM

The final program, SAMSUMM, presents a summary of the study. In an askii file named by the user, the data, final estimates of the parameters, and $\hat{L}_1$, $\hat{L}_2$,..., $\hat{L}_9$, with their estimated standard errors and 95 % confidence limits, are presented. A final graph of $\hat{G}_n(x)$, along with the parameter estimates, is also produced. Figure 6 on pages 134 and 135 presents the output of SAMSUMM for the data of Table 1 (on page 25).

```
*                          INITIAL.SAS                          *;

*****************************************************************;
*   NOTE:  EXISTENCE OF MAXIMUM LIKELIHOOD ESTIMATES IS         *;
* REQUIRES IN ORDER TO USE SAM.  THIS PROGRAM USES A TWO        *;
```

```
* DIMENSIONAL ROBBINS-MONRO PROCEDURE TO GENERATE THE IN- *;
* ITIAL DESIGN LEVELS UNTIL MLEs EXIST. ONCE EXISTENCE IS *;
* OBTAINED, THE PROGRAM NEXTSAM.SAS CAN BE USED TO GENER- *;
* ATE THE NEXT DESIGN LEVELS. ONCE MLEs EXIST, THEY EXIST *;
* AT EACH FUTURE UPDATE (NO NEED TO RERUN THIS PROGRAM)   *;
*                                                         *;
*   IMPORTANT:  THE DATA FILE MUST BE ORDERED SO THAT THE *;
* LAST TWO OBSERVATIONS ARE THE PREVIOUS ESTIMATES OF THE *;
* 20th AND 80th PERCENTILES, RESPECTIVELY.                *;
*                                                         *;
***********************************************************;
****         FILL IN THE FOLLOWING INFORMATION        ****;
****         ---------------------------------        ****;
****         (SEE SAMHELP.DOC FOR MORE DETAILS)       ****;
***********************************************************;
DATA INFO;
*                                                         *;
*   ENTER THE NAME OF THE FILE CONTAINING THE DATA        *;
*   (INCLUDE DIRECTORY OF FILE IE. 'A:\SAMDAT.DAT'        *;
*                                                         *;
           FILENAME DT1 'A:\TEST2.DAT';
*                                                         *;
*   ENTER LOWER AND UPPER BOUNDS FOR THE DESIGN POINTS    *;
*   (IF A . IS ENTERED, THEN A FORMULA BASED ON L20       *;
*   AND L80 WILL BE USED)                                 *;
*                                                         *;
           LG = 10; UG = 400;
*                                                         *;
*   ENTER INITIAL LD20 AND LD80 ESTIMATE                  *;
*                                                         *;
           LD20EST = 60; LD80EST = 100;
*                                                         *;
*   OPTIONAL: ENTER LD50 AND SLOPE ESTIMATE (IF . IS      *;
*             ENTERED FOR THE LD20 AND LD 80 ESTIMATES)   *;
*                                                         *;
           LD50 = .; SLP = .;
*                                                         *;
*******************************************************;
*      NO CHANGES NEEDED BEYOND THIS POINT            *;
*******************************************************;
  IF LD20EST = . THEN LD20EST = LD50 - (1.3863 / SLP);
  IF LD80EST = . THEN LD80EST = LD50 + (1.3863 / SLP);
  SLP = (2*LOG(4)) / (LD80EST - LD20EST);
  LD50EST=(LD20EST+LD80EST)/2;
  IF LG = . THEN LG=(.5)*(LD20EST+LD80EST)-(LOG(50)/SLP);
  IF UG = . THEN UG=(.5)*(LD20EST+LD80EST)+(LOG(50)/SLP);
  CALL SYMPUT('LD20EST',LD20EST); CALL SYMPUT('LG',LG);
  CALL SYMPUT('LD80EST',LD80EST); CALL SYMPUT('UG',UG);
DATA DAT; INFILE DT1;
  KEEP X Y;
  INPUT OBS X Y;
PROC MEANS NOPRINT DATA=DAT N MAX MIN;
  VAR X; OUTPUT OUT=STATS N=NUM MAX=MX MIN=MN;
DATA STATS; SET STATS;
```

```
  LENGTH NUMC $ 10; NUMC = NUM; NUMC = LEFT(NUMC);
  NUMX = 'X'||TRIM(NUMC); NUMY = 'Y'||TRIM(NUMC);
  LNGX = LENGTH(NUMX); LNGC = LENGTH(NUMC);
  CALL SYMPUT('NUMX',NUMX); CALL SYMPUT('NUMY',NUMY);
  CALL SYMPUT('MX',MX); CALL SYMPUT('MN',MN);
  CALL SYMPUT('NUM',NUM); FILE PRINT;
PROC TRANSPOSE DATA=DAT OUT=TRANS;
  VAR X Y;
DATA DATx; SET TRANS;
  ARRAY X {&NUM} X1-&NUMX; KEEP X1-&NUMX DUM;
  IF _NAME_ = 'X'; DUM=1;
  X1=COL1; X2=COL2; X3=COL3; X4=COL4; X5=COL5; X6=COL6;
  X7=COL7; X8=COL8; X9=COL9; X10=COL10; X11=COL11;
  X12=COL12; X13=COL13; X14=COL14; X15=COL15; X16=COL16;
  X17=COL17; X18=COL18; X19=COL19; X20=COL20;
  IF &NUM GT 20 THEN DO;
    X21=COL21; X22=COL22; X23=COL23; X24=COL24; X25=COL25;
    X26=COL26; X27=COL27; X28=COL28; X29=COL29; X30=COL30;
  END;
  IF &NUM GT 30 THEN DO;
    X31=COL31; X32=COL32; X33=COL33; X34=COL34; X35=COL35;
    X36=COL36; X37=COL37; X38=COL38; X39=COL39; X40=COL40;
  END;
  IF &NUM GT 40 THEN DO;
    X41=COL41; X42=COL42; X43=COL43; X44=COL44; X45=COL45;
    X46=COL46; X47=COL47; X48=COL48; X49=COL49; X50=COL50;
  END;
  IF &NUM GT 50 THEN DO;
    X51=COL51; X52=COL52; X53=COL53; X54=COL54; X55=COL55;
    X56=COL56; X57=COL57; X58=COL58; X59=COL59; X60=COL60;
  END;
DATA DATY; SET TRANS;
  ARRAY Y {&NUM} Y1-&NUMY; KEEP Y1-&NUMY DUM;
  IF _NAME_ = 'Y'; DUM=1;
  Y1=COL1; Y2=COL2; Y3=COL3; Y4=COL4; Y5=COL5; Y6=COL6;
  Y7=COL7; Y8=COL8; Y9=COL9; Y10=COL10; Y11=COL11;
  Y12=COL12; Y13=COL13; Y14=COL14; Y15=COL15; Y16=COL16;
  Y17=COL17; Y18=COL18; Y19=COL19; Y20=COL20;
  IF &NUM GT 20 THEN DO;
    Y21=COL21; Y22=COL22; Y23=COL23; Y24=COL24; Y25=COL25;
    Y26=COL26; Y27=COL27; Y28=COL28; Y29=COL29; Y30=COL30;
  END;
  IF &NUM GT 30 THEN DO;
    Y31=COL31; Y32=COL32; Y33=COL33; Y34=COL34; Y35=COL35;
    Y36=COL36; Y37=COL37; Y38=COL38; Y39=COL39; Y40=COL40;
  END;
  IF &NUM GT 40 THEN DO;
    Y41=COL41; Y42=COL42; Y43=COL43; Y44=COL44; Y45=COL45;
    Y46=COL46; Y47=COL47; Y48=COL48; Y49=COL49; Y50=COL50;
  END;
  IF &NUM GT 50 THEN DO;
    Y51=COL51; Y52=COL52; Y53=COL53; Y54=COL54; Y55=COL55;
    Y56=COL56; Y57=COL57; Y58=COL58; Y59=COL59; Y60=COL60;
  END;
```

```
DATA DAT; MERGE DATX DATY;
  BY DUM; DROP DUM;
*;
DATA ONE; SET DAT;
**************************************************************;
***                 INITIALIZE VARIABLES
**************************************************************;
  N = &NUM; CHK=0; DNE=0;
  ARRAY X {&NUM} X1-&NUMX; ARRAY Y {&NUM} Y1-&NUMY;
  IF N LE 3 THEN GO TO FEI;
    ALPHA = .25; BETA = .5;
**************************************************************;
***             CHECK SILVAPULLES CONDITIONS
**************************************************************;
    X0MIN = 1000; X1MIN = 1000;
    X0MAX = -1000; X1MAX = -1000;
    DO I = 1 TO N;
      IF Y{I} = 0 THEN DO;
        IF X{I} GT X0MAX THEN X0MAX = X{I};
        IF X{I} LT X0MIN THEN X0MIN = X{I}; END;
      IF Y{I} = 1 THEN DO;
        IF X{I} GT X1MAX THEN X1MAX = X{I};
        IF X{I} LT X1MIN THEN X1MIN = X{I}; END;
    END;
***                 CONDITION 1                        ***;
    IF X1MAX GT X1MIN AND X0MIN LT X0MAX AND X1MAX GT X0MIN
       AND X1MIN LT X0MAX THEN CHK = 1;
***                 CONDITION 2                        ***;
    IF X0MIN = X0MAX AND X1MIN LT X0MIN AND X0MAX LT X1MAX
       THEN CHK = 1;
***                 CONDITION 3                        ***;
    IF X1MIN = X1MAX AND X0MIN LT X1MIN AND X1MAX LT X0MAX
       THEN CHK = 1;
    LENGTH TYP1 $ 20 TYP2 $ 35 TYP3 $ 40 WARN1 $ 35 WARN2 $
           25 SUGX1C $ 15 SUGX2C $ 15 ANDM $ 3 SGXT3 $ 16;
    IF CHK = 1 THEN DO;
**************************************************************;
***                   IF MLEs EXIST                      ***;
***                  RESCALE THE DATA                    ***;
**************************************************************;
      RANG = &MX - &MN; MID = (&MX + &MN) / 2;
      DO I = 1 TO &NUM;
        X{I} = (X{I} - MID)*(8/RANG);
      END;
*;
***                  CALCULATE MLE'S                     ***;
*;
    G1=0; G2=0; FLGDET = 0; FLGNEG = 0; FLG=0;
    DO J = 1 TO 10;
      IF (G1**2 + G2**2) LT .0001 AND J NE 1 THEN GO TO OK1;
      H11 = 0; H12 = 0; H22 = 0;
      G1 = 0; G2 = 0;
      DO I = 1 TO N;
        Z = ALPHA + BETA*X{I};
```

```
        IF ABS(Z) GT 15 THEN DO; PRED=1; GO TO DE3; END;
        PRED = EXP(Z) / (1 + EXP(Z));
DE3:    H11 = H11 - PRED*(1-PRED);
        H12 = H12 - X{I}*PRED*(1-PRED);
        H22 = H22 - X{I}*X{I}*PRED*(1-PRED);
        G1 = G1 + (Y{I} - PRED);
        G2 = G2 + (X{I}*Y{I} - X{I}*PRED);
      END;
      IF (H11*H22) - (H12**2) LT .001 THEN DO;
       PUT 'DET NEAR ZERO'; DET=1; GO TO OK1; END;
      DET = (H11*H22) - (H12**2);
      HINV11 = H22 / DET;
      HINV22 = H11 / DET;
      HINV12 = -(H12 / DET);
      ALPHA = ALPHA - ((HINV11*G1) + (HINV12*G2));
      BETA  = BETA  - ((HINV12*G1) + (HINV22*G2));
    END;
OK1:  FILE PRINT;
    BETA=BETA*8/(&MX - &MN); ALPHA=(-ALPHA/BETA) + ((&MX +
         &MN)/2);
   ALPHA = -ALPHA*BETA;
*;
***              END OF CALCULATE MLE'S                   ***;
*;
        TYP1 = 'MLE EXISTS';
        TYP2 = 'RUN NEXTSAM PROGRAM '; TYP3 = ' ';
        SUGX1 = .; SUGX2 = .;
        WARN1 = ' '; WARN2 = ' ';
        LBSLP = (2.77 / (&LD80EST - &LD20EST)) / 20;
        IF BETA LE LBSLP THEN DO;
          TYP2 = 'WARNING: NEGATIVE OR SMALL SLOPE';
          FLGNEG = 1; GO TO RSC;
        END;
        IF DET = 1 THEN DO;
          TYP2 = 'WARNING: DETERMINANT NEAR ZERO';
          FLGDET = 1; GO TO RSC;
        END;
***            RETURN DATA TO ORIGINAL SCALE              ***;
RSC: DO V=1 TO &NUM;
     X{V} = X{V}*((&MX - &MN)/8) + ((&MX + &MN) / 2);
   END;
   IF FLGNEG=1 OR FLGDET=1 THEN GO TO FEI;
      DNE=2; GO TO SP1; END;
    ELSE DO;
*************************************************************;
***               IF MLEs DO NOT EXIST
*************************************************************;
      PUT 'MLE DOES NOT EXIST'; DNE=1;
*;
* CALCULATE DESIGN POINTS USING TWO-DIMENSIONAL RM PROCEDURE
*;
FEI:  PUT 'ENTERED FEI';
    IF X{N} LE X{N-1} THEN DO;
      SUGX1 = .; SUGX2 = .; PUT 'ERROR IN DATA';
```

```
      WARN1 = 'ERROR IN ORDER OF DATA';
      WARN2 = 'ESTIMATE OF L20 > L80'; FLG = 1;
    GO TO ERR; END;
    ANHAT = (X{N} - X{N-1}) / (.4436*N);
    SUGX1 = X{N-1} - ANHAT*(Y{N-1} - .2);
    SUGX2 = X{N} - ANHAT*(Y{N} - .8);
    IF SUGX1 LT &LG THEN SUGX1 = &LG;
    IF SUGX2 GT &UG THEN SUGX2 = &UG;
    IF N LE 3 THEN DO;
        TYP1 = 'MLE DOES NOT EXIST';
        TYP2 = ' ';
        TYP3 = '2-DIMENSIONAL RM DESIGN LEVELS:';
    GO TO SP1; END;
*;
***          FIND REASON FOR NONEXISTENCE         ***;
*;
      IF X0MAX = -1000 THEN DO;   * NO ZERO RESPONSES *;
*PUT 'ENTERED 1';
        TYP1 = 'MLE DOES NOT EXIST';
        TYP2 = 'ALL RESPONSES ARE ONES';
        TYP3 = '2-DIMENSIONAL RM DESIGN LEVELS:';
      END;
      IF X1MAX = -1000 THEN DO; * ALL ZERO RESPONSES *;
*PUT 'ENTERED 2';
        TYP1 = 'MLE DOES NOT EXIST';
        TYP2 = 'ALL RESPONSES ARE ZEROES';
        TYP3 = '2-DIMENSIONAL RM DESIGN LEVELS:';
      END;
*          NO OVERLAP IN THE RESPONSES                       *;
      IF X0MAX LE X1MIN OR X0MIN GE X1MAX THEN DO;
        IF X0MAX=-1000 OR X1MAX=-1000 THEN GO TO SP1;
*PUT 'ENTERED 3' X0MIN X0MAX X1MIN X1MAX;
        TYP1 = 'MLE DOES NOT EXIST';
        TYP2 = 'NO OVERLAP IN THE RESPONSES';
        TYP3 = '2-DIMENSIONAL RM DESIGN LEVELS:';
      END;
    END;
***            END:  MLEs DO NOT EXIST SECTION              ***;
***;
***                    PREPARE MESSAGE                      ***;
***;
SP1:  SUGX1 = ROUND(SUGX1,.01); SUGX2 = ROUND(SUGX2,.01);
     SUGX1C = SUGX1;  SUGX2C = SUGX2 ; ANDM='AND';
     SUGX1C = LEFT(SUGX1C);  SUGX2C = LEFT(SUGX2C);
     IF DNE=2 THEN SGXT3 = ' ';
       ELSE SGXT3 = TRIM(SUGX1C)||' '||ANDM||' '||SUGX2C;
ERR: IF FLG = 1 THEN DO;
      TYP1 = 'MLE DOES NOT EXIST';
      TYP2 = 'UNABLE TO CALCULATE LEVELS';
    END;
    CALL SYMPUT('TYP1',TYP1);  CALL SYMPUT('TYP2',TYP2);
    CALL SYMPUT('TYP3',TYP3);  CALL SYMPUT('SGXT3',SGXT3);
    CALL SYMPUT('WARN1',WARN1); CALL SYMPUT('WARN2',WARN2);
GOPTIONS DEVICE = HERCULES;
```

```
PROC GSLIDE;
  TITLE J=C H=.55 IN 'SAM';
  NOTE J=L H=.35 IN ' ';
  NOTE J=L H=.35 IN "&TYP1";
  NOTE J=L H=.15 IN ' ';
  NOTE J=L H=.35 IN "&TYP2";
  NOTE J=L H=.15 IN ' ';
  NOTE J=L H=.35 IN "&TYP3";
  NOTE J=L H=.15 IN ' ';
  NOTE J=L H=.35 IN "&SGXT3";
  NOTE J=L H=.15 IN ' ';
  NOTE J=L H=.25 IN "&WARN1";
  NOTE J=L H=.1 IN ' ';
  NOTE J=L H=.25 IN "&WARN2";
RUN; QUIT;



*                      MLECHECK                              *;

*   NOTE :  EXISTENCE OF MAXIMUM LIKELIHOOD ESTIMATES IS  *;
* REQUIRED IN ORDER TO USE SAM.  THIS PROGRAM CHECKS FOR  *;
* EXISTENCE OF MLEs.  If MLEs EXIST, THEN THE PROGRAM     *;
* NEXTSAM.SAS CAN BE USED TO GENERATE THE NEXT DESIGN     *;
* LEVELS. IF MLEs EXIST AT A GIVEN UPDATE THEN THEY EXIST *;
* AT EACH FUTURE UPDATE (NO NEED TO RERUN THIS PROGRAM).  *;
*                                                         *;
**********************************************************;
****        FILL IN THE FOLLOWING INFORMATION        ****;
****        --------------------------------         ****;
****        (SEE SAMHELP.DOC FOR MORE DETAILS)       ****;
**********************************************************;
DATA UNO;
*   ENTER THE NAME OF THE FILE CONTAINING THE DATA        *;
*   (INCLUDE DIRECTORY OF FILE IE. 'A:\SAMDAT.DAT'        *;
           FILENAME DT1 'A:\TEST.DAT';
*   ENTER LOWER AND UPPER BOUND FOR THE DESIGN POINTS     *;
*   (IF A . IS ENTERED, THEN A FORMULA BASED ON L20       *;
*   AND L80 WILL BE USED)                                 *;
           LG = 0; UG = 150;
*   ENTER INITIAL LD20 AND LD80 ESTIMATE                  *;
           LD20EST = 60; LD80EST = 100;
*                                                         *;
*******************************************************;
*       NO CHANGES NEEDED BEYOND THIS POINT           *;
*******************************************************;
  SLPEST = (2*LOG(4)) / (LD80EST - LD20EST);
  IF LG = . THEN LG = (.5)*(LD20EST+LD80EST) -
(LOG(50)/SLPEST);
  IF UG = . THEN UG = (.5)*(LD20EST+LD80EST) +
(LOG(50)/SLPEST);
  CALL SYMPUT('LG',LG);  CALL SYMPUT('UG',UG);
```

```
  CALL SYMPUT('LD20EST',LD20EST);  CALL
SYMPUT('LD80EST',LD80EST);
  FILE PRINT;
DATA DAT; INFILE DT1;
  KEEP X Y;
  INPUT OBS X Y;
PROC MEANS NOPRINT DATA=DAT N MAX MIN;
  VAR X; OUTPUT OUT=STATS N=NUM MAX=MX MIN=MN;
DATA STATS; SET STATS;
  LENGTH NUMC $ 10; NUMC = NUM; NUMC = LEFT(NUMC);
  NUMX = 'X'||TRIM(NUMC); NUMY = 'Y'||TRIM(NUMC);
  LNGX = LENGTH(NUMX); LNGC = LENGTH(NUMC);
  CALL SYMPUT('NUMX',NUMX); CALL SYMPUT('NUMY',NUMY);
  CALL SYMPUT('MX',MX); CALL SYMPUT('MN',MN);
  CALL SYMPUT('NUM',NUM);
PROC TRANSPOSE DATA=DAT OUT=TRANS;
  VAR X Y;
DATA DATX; SET TRANS;
  ARRAY X {&NUM} X1-&NUMX; KEEP X1-&NUMX DUM;
  IF _NAME_ = 'X'; DUM=1;
  X1=COL1; X2=COL2; X3=COL3; X4=COL4; X5=COL5; X6=COL6;
  X7=COL7; X8=COL8; X9=COL9; X10=COL10; X11=COL11;
  X12=COL12; X13=COL13; X14=COL14; X15=COL15; X16=COL16;
  X17=COL17; X18=COL18; X19=COL19; X20=COL20;
  IF &NUM GT 20 THEN DO;
    X21=COL21; X22=COL22; X23=COL23; X24=COL24; X25=COL25;
    X26=COL26; X27=COL27; X28=COL28; X29=COL29; X30=COL30;
  END;
  IF &NUM GT 30 THEN DO;
    X31=COL31; X32=COL32; X33=COL33; X34=COL34; X35=COL35;
    X36=COL36; X37=COL37; X38=COL38; X39=COL39; X40=COL40;
  END;
  IF &NUM GT 40 THEN DO;
    X41=COL41; X42=COL42; X43=COL43; X44=COL44; X45=COL45;
    X46=COL46; X47=COL47; X48=COL48; X49=COL49; X50=COL50;
  END;
  IF &NUM GT 50 THEN DO;
    X51=COL51; X52=COL52; X53=COL53; X54=COL54; X55=COL55;
    X56=COL56; X57=COL57; X58=COL58; X59=COL59; X60=COL60;
  END;
DATA DATY; SET TRANS;
  ARRAY Y {&NUM} y1-&NUMY; KEEP Y1-&NUMY DUM;
  IF _NAME_ = 'Y'; DUM=1;
  Y1=COL1; Y2=COL2; Y3=COL3; Y4=COL4; Y5=COL5; Y6=COL6;
  Y7=COL7; Y8=COL8; Y9=COL9; Y10=COL10; Y11=COL11;
  Y12=COL12; Y13=COL13; Y14=COL14; Y15=COL15; Y16=COL16;
  Y17=COL17; Y18=COL18; Y19=COL19; Y20=COL20;
  IF &NUM GT 20 THEN DO;
    Y21=COL21; Y22=COL22; Y23=COL23; Y24=COL24; Y25=COL25;
    Y26=COL26; Y27=COL27; Y28=COL28; Y29=COL29; Y30=COL30;
  END;
  IF &NUM GT 30 THEN DO;
    Y31=COL31; Y32=COL32; Y33=COL33; Y34=COL34; Y35=COL35;
    Y36=COL36; Y37=COL37; Y38=COL38; Y39=COL39; Y40=COL40;
```

```
  END;
  IF &NUM GT 40 THEN DO;
    Y41=COL41; Y42=COL42; Y43=COL43; Y44=COL44; Y45=COL45;
    Y46=COL46; Y47=COL47; Y48=COL48; Y49=COL49; Y50=COL50;
  END;
  IF &NUM GT 50 THEN DO;
    Y51=COL51; Y52=COL52; Y53=COL53; Y54=COL54; Y55=COL55;
    Y56=COL56; Y57=COL57; Y58=COL58; Y59=COL59; Y60=COL60;
END;
DATA DAT; MERGE DATX DATY;
  BY DUM; DROP DUM;
***                  RESCALE THE DATA                    ***;
DATA DAT; SET DAT;
  ARRAY X {&NUM} X1-&NUMX; ARRAY Y {&NUM} Y1-&NUMY;
  RANG = &MX - &MN; MID = (&MX + &MN) / 2;
  DO I = 1 TO &NUM;
    X{I} = (X{I} - MID)*(8/RANG);
  END;
*;
DATA ONE; SET DAT;
************************************************************;
***                  INITIALIZE VARIABLES
************************************************************;
  N = &NUM; CHK=0; DNE=0;
FILE PRINT;
  ARRAY X {&NUM} X1-&NUMX; ARRAY Y {&NUM} Y1-&NUMY;
    ALPHA = .25; BETA = .5;
************************************************************
***              CHECK SILVAPULLES CONDITIONS
************************************************************
    X0MIN = 1000; X1MIN = 1000;
    X0MAX = -1000; X1MAX = -1000;
    DO I = 1 TO N;
      IF Y{I} = 0 THEN DO;
        IF X{I} GT X0MAX THEN X0MAX = X{I};
        IF X{I} LT X0MIN THEN X0MIN = X{I}; END;
      IF Y{I} = 1 THEN DO;
        IF X{I} GT X1MAX THEN X1MAX = X{I};
        IF X{I} LT X1MIN THEN X1MIN = X{I}; END;
    END;
***                  CONDITION 1                         ***;
    IF X1MAX GT X1MIN AND X0MIN LT X0MAX AND X1MAX GT X0MIN
       AND X1MIN LT X0MAX THEN CHK = 1;
***                  CONDITION 2                         ***;
    IF X0MIN = X0MAX AND X1MIN LT X0MIN AND X0MAX LT X1MAX
       THEN CHK = 1;
***                  CONDITION 3                         ***;
    IF X1MIN = X1MAX AND X0MIN LT X1MIN AND X1MAX LT X0MAX
       THEN CHK = 1;
    LENGTH TYP1 $ 20 TYP2 $ 30 TYP3 $ 40 WARN1 $ 35 WARN2 $
           25 SUGX1C $ 15 SUGX2C $ 15 ANDM $ 3;
    IF CHK = 1 THEN DO;
************************************************************;
***                    CALCULATE MLE'S
```

```
**********************************************************;
    G1=0; G2=0;
    DO J = 1 TO 10;
      IF (G1**2 + G2**2) LT .0001 AND J NE 1 THEN GO TO OK1;
      H11 = 0; H12 = 0; H22 = 0;
      G1 = 0; G2 = 0;
      DO I = 1 TO N;
        Z = ALPHA + BETA*X{I};
        IF ABS(Z) GT 15 THEN DO; PRED=1; GO TO DE3; END;
        PRED = EXP(Z) / (1 + EXP(Z));
DE3:    H11 = H11 - PRED*(1-PRED);
        H12 = H12 - X{I}*PRED*(1-PRED);
        H22 = H22 - X{I}*X{I}*PRED*(1-PRED);
        G1 = G1 + (Y{I} - PRED);
        G2 = G2 + (X{I}*Y{I} - X{I}*PRED);
      END;
      IF (H11*H22) - (H12**2) LT .001 THEN DO;
       PUT 'DET NEAR ZERO'; DNE=1; GO TO OK1; END;
      DET = (H11*H22) - (H12**2);
      HINV11 = H22 / DET;
      HINV22 = H11 / DET;
      HINV12 = -(H12 / DET);
      ALPHA = ALPHA - ((HINV11*G1) + (HINV12*G2));
      BETA  = BETA  - ((HINV12*G1) + (HINV22*G2));
    END;
OK1:  FILE PRINT;
    BETA=BETA*8/(&MX - &MN); ALPHA=(-ALPHA/BETA) + ((&MX +
        &MN)/2);
    ALPHA = -ALPHA*BETA;
*;
**********************************************************;
***              END OF CALCULATE MLE'S
**********************************************************;
*;
        TYP1 = 'MLE EXISTS';
        TYP2 = 'RUN NEXTSAM PROGRAM '; TYP3 = ' ';
        SUGX1 = .; SUGX2 = .;
        WARN1 = ' '; WARN2 = ' ';
        IF BETA LE 0 THEN DO;
          TYP2 = 'WARNING: NEGATIVE OR ZERO SLOPE';
          TYP3 = 'WE RECOMMEND THE FOLLOWING DESIGN POINTS';
          SUGX1 = &LD20EST; SUGX2 = &LD80EST;
        END;
        IF DNE = 1 THEN DO;
          TYP2 = 'WARNING: DETERMINANT NEAR ZERO';
          TYP3 = 'WE RECOMMEND THE FOLLOWING DESIGN POINTS';
          SUGX1 = &LD20EST; SUGX2 = &LD80EST;
        END;
        GO TO SP1; END;
    ELSE DO;
      PUT 'MLE DOES NOT EXIST'; DNE=1;
***         RETURN X0MIN  ETC. TO ORIGINAL SCALE        ***;
/*
   DO V=1 TO &NUM;
```

```
    X{V} = X{V}*((&MX - &MN)/8) + ((&MX + &MN) / 2);
   END;
*/
   IF X0MIN NE 1000 THEN
     X0MIN = X0MIN*((&MX - &MN)/8) + ((&MX + &MN) / 2);
   IF X0MAX NE -1000 THEN
     X0MAX = X0MAX*((&MX - &MN)/8) + ((&MX + &MN) / 2);
   IF X1MIN NE 1000 THEN
     X1MIN = X1MIN*((&MX - &MN)/8) + ((&MX + &MN) / 2);
   IF X1MAX NE -1000 THEN
     X1MAX = X1MAX*((&MX - &MN)/8) + ((&MX + &MN) / 2);
***      END - RETURN X0MIN  ETC. TO ORIGINAL SCALE    ***;
************                                   ***********;
* CALCULATE SUGGESTED DESIGN POINTS FOR WHEN NO MLE EXISTS*;
************                                   ***********;
      IF X0MAX = -1000 THEN DO;   * NO ZERO RESPONSES *;
        TYP1 = 'MLE DOES NOT EXIST';
        TYP2 = 'ALL RESPONSES ARE ONES';
        TYP3 = 'WE RECOMMEND THE FOLLOWING DESIGN POINTS';
        SUGX1 = X1MIN - (.5)*(X1MIN - &LG);
        SUGX2 = X1MIN - (.1)*(X1MIN - &LG);
        IF X1MIN LE &LG THEN DO;
          WARN1 = 'THE LOWER BOUND MAY BE TOO LARGE';
          WARN2 = 'ENTER A NEW LOWER BOUND';
          SUGX1 = &LG - (1/3)*(&LD80EST - &LD20EST);
          SUGX2 = &LG;
        END; END;
      IF X1MAX = -1000 THEN DO;   * ALL ZERO RESPONSES *;
        TYP1 = 'MLE DOES NOT EXIST';
        TYP2 = 'ALL RESPONSES ARE ZEROES';
        TYP3 = 'WE RECOMMEND THE FOLLOWING DESIGN POINTS';
        SUGX1 = X0MAX + (.1)*(&UG - X0MAX);
        SUGX2 = X0MAX + (1/2)*(&UG - X0MAX);
        IF X0MAX GE &UG THEN DO;
          WARN1 = 'THE UPPER BOUND MAY BE TOO SMALL';
          WARN2 = 'ENTER A NEW UPPER BOUND';
          SUGX1 = &UG;
          SUGX2 = &UG + (1/3)*(&LD80EST - &LD20EST);
        END; END;
      IF X0MAX LE X1MIN THEN DO;  * NO OVERLAP IN RESPONSES
                                      (1s HIGHER) *;
        IF X0MAX=-1000 OR X1MAX=-1000 THEN GO TO SP1;
        SUGX1 = X0MAX + (1/3)*(X1MIN - X0MAX);
        SUGX2 = X0MAX + (2/3)*(X1MIN - X0MAX);
        IF X0MAX = X1MIN THEN DO;
          SUGX1 = X0MAX - (.1)*(&LD80EST - &LD20EST);
          SUGX2 = X0MAX + (.1)*(&LD80EST - &LD20EST);
        END;
        TYP1 = 'MLE DOES NOT EXIST';
        TYP2 = 'NO OVERLAP IN THE RESPONSES';
        TYP3 = 'WE RECOMMEND THE FOLLOWING DESIGN POINTS';
      END;
      IF X0MIN GE X1MAX THEN DO;* NO OVERLAP IN RESPONSES
                                    (0s HIGHER) *;
```

```
        IF X0MAX=-1000 OR X1MAX=-1000 THEN GO TO SP1;
        SUGX1 = X1MAX + (1/3)*(X0MIN - X1MAX);
        SUGX2 = X1MAX + (2/3)*(X0MIN - X1MAX);
        IF X0MIN = X1MAX THEN DO;
          SUGX1 = X1MAX - (.1)*(&LD80EST - &LD20EST);
          SUGX2 = X1MAX + (.1)*(&LD80EST - &LD20EST);
        END;
        TYP1 = 'MLE DOES NOT EXIST';
        TYP2 = 'NO OVERLAP IN THE RESPONSES';
        TYP3 = 'WE RECOMMEND THE FOLLOWING DESIGN POINTS';
      END;
    END;
SP1:  SUGX1 = ROUND(SUGX1,.01); SUGX2 = ROUND(SUGX2,.01);
    SUGX1C = SUGX1;  SUGX2C = SUGX2 ; ANDM='AND';
    SUGX1C = LEFT(SUGX1C);  SUGX2C = LEFT(SUGX2C);
    SGXT3 = TRIM(SUGX1C)||' '||ANDM||' '||SUGX2C;
    IF DNE NE 1 AND BETA GT 0 THEN SGXT3 = ' ';
    CALL SYMPUT('TYP1',TYP1);  CALL SYMPUT('TYP2',TYP2);
    CALL SYMPUT('TYP3',TYP3);  CALL SYMPUT('SGXT3',SGXT3);
    CALL SYMPUT('WARN1',WARN1); CALL SYMPUT('WARN2',WARN2);
GOPTIONS DEVICE = HERCULES;
PROC GSLIDE;
  TITLE J=C H=.55 IN 'SAM';
  NOTE J=L H=.35 IN ' ';
  NOTE J=L H=.35 IN "&TYP1";
  NOTE J=L H=.15 IN ' ';
  NOTE J=L H=.35 IN "&TYP2";
  NOTE J=L H=.15 IN ' ';
  NOTE J=L H=.35 IN "&TYP3";
  NOTE J=L H=.15 IN ' ';
  NOTE J=L H=.35 IN "&SGXT3";
  NOTE J=L H=.15 IN ' ';
  NOTE J=L H=.25 IN "&WARN1";
  NOTE J=L H=.1 IN ' ';
  NOTE J=L H=.25 IN "&WARN2";
RUN; QUIT;



*                          NEXTSAM                          *;

* NOTE:  THIS PROGRAM GENERATES THE NEXT DESIGN POINTS  *;
* USING SAM GIVEN A SET OF DATA FOR WHICH MAXIMUM       *;
* LIKELIHOOD ESTIMATES EXIST.  THE PROGRAM MLECHECK.SAS *;
* SHOULD BE RUN TO CHECK FOR MLEs BEFORE USING THIS     *;
* PROGRAM.                                              *;
*                                                       *;
DATA UNO;
*******************************************************;
*       FILL IN THE FOLLOWING INFORMATION             *;
*******************************************************;
*  ENTER NAME OF FILE CONTAINING THE DATA             *;
*   (INCLUDE PATH NAME ie. 'C:\SAMDAT.DAT')           *;
```

```
        filename dat1 'A:\TEST3.DAT';
*  ENTER LOWER AND UPPER BOUND FOR THE DESIGN POINTS   *;
*   (WE SUGGEST A FORMULA BASED ON L20 L80)            *;
*   (THESE WILL BE USED IF YOU LEAVE THE SPACES BLANK) *;
         LG = 0; UG = 300;
*  ENTER LD20 AND LD80 ESTIMATE                        *;
         LD20EST = 150; LD80EST = 200;
*  ARE THE LAST TWO POINTS ESTIMATES GENERATED BY SAM  *;
*   (ENTER 1 SAM HAS BEEN PREVIOUSLY RUN, 0 OTHERWISE  *;
         FRST = 0;
*******************************************************;
*       NO CHANGES NEEDED BEYOND THIS POINT            *;
*******************************************************;
CALL SYMPUT('FRST',FRST);   CALL SYMPUT('LG',LG);
CALL SYMPUT('UG',UG);   CALL SYMPUT('LD20EST',LD20EST);
CALL SYMPUT('LD80EST',LD80EST);
DATA DAT;  INFILE DAT1;  KEEP X Y;
  INPUT OBS X Y;
PROC MEANS NOPRINT DATA=DAT N MAX MIN;
  VAR X; OUTPUT OUT=STATS N=NUM MAX=MX MIN=MN;
DATA STATS; SET STATS;
  LENGTH NUMC $8; NUMC = NUM; NUMC = LEFT(NUMC);
  NUMX = 'X'||TRIM(NUMC); NUMY = 'Y'||TRIM(NUMC);
  CALL SYMPUT('NUMX',NUMX); CALL SYMPUT('NUMY',NUMY);
  CALL SYMPUT('MX',MX); CALL SYMPUT('MN',MN);
  CALL SYMPUT('NUM',NUM);
PROC TRANSPOSE DATA=DAT OUT=TRANS;
  VAR X Y;
DATA DATX; SET TRANS;
  ARRAY X {&NUM} X1-&NUMX; KEEP X1-&NUMX DUM;
  IF _NAME_ = 'X'; DUM=1;
  X1=COL1; X2=COL2; X3=COL3; X4=COL4; X5=COL5; X6=COL6;
  X7=COL7; X8=COL8; X9=COL9; X10=COL10; X11=COL11;
  X12=COL12; X13=COL13; X14=COL14; X15=COL15; X16=COL16;
  X17=COL17; X18=COL18;
  IF &NUM GT 18 THEN DO;
    X19=COL19; X20=COL20; X21=COL21; X22=COL22; X23=COL23;
    X24=COL24; X25=COL25; X26=COL26; X27=COL27; X28=COL28;
    X29=COL29; X30=COL30;
  END;
  IF &NUM GT 30 THEN DO;
    X31=COL31; X32=COL32; X33=COL33; X34=COL34; X35=COL35;
    X36=COL36; X37=COL37; X38=COL38; X39=COL39; X40=COL40;
  END;
  IF &NUM GT 40 THEN DO;
    X41=COL41; X42=COL42; X43=COL43; X44=COL44; X45=COL45;
    X46=COL46; X47=COL47; X48=COL48; X49=COL49; X50=COL50;
  END;
  IF &NUM GT 50 THEN DO;
    X51=COL51; X52=COL52; X53=COL53; X54=COL54; X55=COL55;
    X56=COL56; X57=COL57; X58=COL58; X59=COL59; X60=COL60;
  END;
DATA DATY; SET TRANS;
  ARRAY Y {&NUM} Y1-&NUMY; KEEP Y1-&NUMY DUM;
```

```
  IF _NAME_ = 'Y'; DUM=1;
  Y1=COL1; Y2=COL2; Y3=COL3; Y4=COL4; Y5=COL5; Y6=COL6;
  Y7=COL7; Y8=COL8; Y9=COL9; Y10=COL10; Y11=COL11;
  Y12=COL12; Y13=COL13; Y14=COL14; Y15=COL15; Y16=COL16;
  Y17=COL17; Y18=COL18;
  IF &NUM GT 18 THEN DO;
    Y19=COL19; Y20=COL20; Y21=COL21; Y22=COL22; Y23=COL23;
    Y24=COL24; Y25=COL25; Y26=COL26; Y27=COL27; Y28=COL28;
    Y29=COL29; Y30=COL30;
  END;
  IF &NUM GT 30 THEN DO;
    Y31=COL31; Y32=COL32; Y33=COL33; Y34=COL34; Y35=COL35;
    Y36=COL36; Y37=COL37; Y38=COL38; Y39=COL39; Y40=COL40;
  END;
  IF &NUM GT 40 THEN DO;
    Y41=COL41; Y42=COL42; Y43=COL43; Y44=COL44; Y45=COL45;
    Y46=COL46; Y47=COL47; Y48=COL48; Y49=COL49; Y50=COL50;
  END;
  IF &NUM GT 50 THEN DO;
    Y51=COL51; Y52=COL52; Y53=COL53; Y54=COL54; Y55=COL55;
    Y56=COL56; Y57=COL57; Y58=COL58; Y59=COL59; Y60=COL60;
  END;
DATA DAT MERGE DATX DATY;
  BY DUM; DROP DUM;
***                RESCALE THE DATA                      ***;
DATA DAT; SET DAT;
  ARRAY X {&NUM} X1-&NUMX; ARRAY Y {&NUM} Y1-&NUMY;
  RANG = &MX - &MN; MID = (&MX + &MN) / 2;
  DO I = 1 TO &NUM;
    X{I} = (X{I} - MID)*(8 / RANG);
  END;
***                                                      ***;
DATA ONE; SET DAT;
**********************************************************
***                 INITIALIZE VARIABLES
**********************************************************
****;
  N = &NUM; CHK=0; DNE=0; FILE PRINT;
  UB=50*((&LD80EST - &LD20EST)/2.77)*(8/(&MX - &MN));
  SLPBD = (2.77 / (&LD80EST - &LD20EST)) / 25;
  ARRAY X {&NUM} X1-&NUMX; ARRAY Y {&NUM} Y1-&NUMY;
    ALPHA = .25; BETA = .5;
**********************************************************
***;
***             CHECK SILVAPULLE'S CONDITIONS
***;
**********************************************************
***;
    X0MIN = 50; X1MIN = 50;
    X0MAX = -50; X1MAX = -50;
    DO I = 1 TO N;
      IF Y{I} = 0 THEN DO;
        IF X{I} GT X0MAX THEN X0MAX = X{I};
        IF X{I} LT X0MIN THEN X0MIN = X{I}; END;
```

```
      IF Y{I} = 1 THEN DO;
        IF X{I} GT X1MAX THEN X1MAX = X{I};
        IF X{I} LT X1MIN THEN X1MIN = X{I}; END;
    END;
***                 CONDITION 1
***;
    IF X1MAX GT X1MIN AND X0MIN LT X0MAX AND X1MAX GT
X0MIN AND
       X1MIN LT X0MAX THEN CHK = 1;
***                 CONDITION 2
***;
    IF X0MIN = X0MAX AND X1MIN LT X0MIN AND X0MAX LT X1MAX
       THEN CHK = 1;
***                 CONDITION 3
***;
    IF X1MIN = X1MAX AND X0MIN LT X1MIN AND X1MAX LT X0MAX
       THEN CHK = 1;
    IF CHK = 0 THEN DO;
      PUT 'MLE DOES NOT EXIST'; DNE=1;
 *    ENDSAS; GO TO OK2;
    END;
**********************************************************
***;
***                   CALCULATE MLE'S
***;
**********************************************************
***;
    G1=0; G2=0;
    DO J = 1 TO 10;
      IF (G1**2 + G2**2) LT .0001 AND J NE 1 THEN GO TO
OK1;
      H11 = 0; H12 = 0; H22 = 0;
      G1 = 0; G2 = 0;
      DO I = 1 TO N;
        Z = ALPHA + BETA*X{I};
        IF ABS(Z) GT 15 THEN DO; PRED=1; GO TO DE3; END;
        PRED = EXP(Z) / (1 + EXP(Z));
DE3:    H11 = H11 - PRED*(1-PRED);
        H12 = H12 - X{I}*PRED*(1-PRED);
        H22 = H22 - X{I}*X{I}*PRED*(1-PRED);
        G1 = G1 + (Y{I} - PRED);
        G2 = G2 + (X{I}*Y{I} - X{I}*PRED);
      END;
      IF (H11*H22) - (H12**2) LT .001 THEN DO;
       PUT 'DET NEAR ZERO'; DNE=1; GO TO OK; END;
      DET = (H11*H22) - (H12**2);
      HINV11 = H22 / DET;
      HINV22 = H11 / DET;
      HINV12 = -(H12 / DET);
      ALPHA = ALPHA - ((HINV11*G1) + (HINV12*G2));
      BETA  = BETA  - ((HINV12*G1) + (HINV22*G2));
    END;
***            RESCALE THE DATA BACK TO NORMAL
***;
```

```
OK1: BETA=BETA*8/(&MX - &MN); ALPHA=(-ALPHA / BETA)+((&MX
          + &MN)/2);
 ALPHA = -ALPHA*BETA;
 DO V = 1 TO &NUM;
   X{V} = X{V}*((&MX - &MN)/8) + ((&MX + &MN)/2);
 END;
* USE FRST TO DECIDE HOW TO CALCULATE NEXT DESIGN POINTS
*;
* IE.  USE DSTAR OR JUST THE MLES WITH NO STEP SIZE BOUND
*;
**********************************************************
***;
***                CALCULATE STARTING VALUES
***                NOT USING STEP SIZE BOUNDS
***;
**********************************************************
***;
   IF &FRST=0 THEN DO;
   IF BETA LE SLPBD THEN BETA = SLPBD;
*  IF BETA GT &UBBETA THEN BETA = &UBBETA;
  XSMF50=-ALPHA / BETA;
  XSMF80 = XSMF50 + (LOG(4)/BETA);
  XSMF20 = XSMF50 - (LOG(4)/BETA);
  IF XSMF80 GT &UG THEN XSMF80 = &UG;
  IF XSMF20 LT &LG THEN XSMF20 = &LG;
  IF XSMF50 GT &UG THEN XSMF50=&UG; IF XSMF50 LT &LG THEN
XSMF50=&LG;
  GO TO OK;
END;
**********************************************************
***;
***                USING STEP SIZE BOUNDS
***;
**********************************************************
***;
     IF &FRST=1 THEN DO;
     IF BETA LE SLPBD THEN BETA = SLPBD;
*     IF BETA GT &UBBETA THEN BETA = &UBBETA;
     XSMF20 = (LOG(.25) - ALPHA)/BETA;
     XSMF80 = (LOG(4) - ALPHA)/BETA;
 DSTAR=(X{N-1} - XSMF20)*N/(Y{N-1} - .2);
 DSTAR2 = (X{N} - XSMF80)*N / (Y{N} - .8);
     IF DSTAR GT UB THEN DSTAR=UB;
     IF DSTAR LT (-UB) THEN DSTAR=-UB;
     IF DSTAR2 GT UB THEN DSTAR=UB;
     IF DSTAR2 LT (-UB) THEN DSTAR=-UB;
     IF BETA LT 0 THEN DO; DSTAR=UB;DSTAR2=UB; END;
**********************************************************
***;
***                   GENERATE NEXT X'S
***;
**********************************************************
***;
     XSMF20 = X{N-1}-(DSTAR/N)*(Y{N-1}-.2);
```

```
     XSMF80 = X{N}-(DSTAR2/N)*(Y{N}-.8);
     XSMF50 = -ALPHA/BETA;
     IF XSMF20 GT &UG THEN XSMF20 = &UG;
     IF XSMF80 GT &UG THEN XSMF80 = &UG;
     IF XSMF20 LT &LG THEN XSMF20 = &LG;
     IF XSMF80 LT &LG THEN XSMF80 = &LG;
   END;
OK:  R1 = ROUND(XSMF50,.01); R2 = ROUND(XSMF20,.01);
R3 = ROUND(XSMF80,.01); ALPHAN = -ALPHA / BETA;
LG2 = ALPHAN - (LOG(75) / BETA); UG2 = ALPHAN + (LOG(75) /
BETA);
BYN = (UG2 - LG2) / 300; OFS = (UG2 - LG2) / 75;
CALL SYMPUT('BYN',BYN);  CALL SYMPUT('OFS',OFS);
CALL SYMPUT('L50',XSMF50);  CALL SYMPUT('L50RND',R1);
CALL SYMPUT('LG2',LG2);  CALL SYMPUT('UG2',UG2);
CALL SYMPUT('L20',XSMF20);  CALL SYMPUT('L80',XSMF80);
CALL SYMPUT('L20RND',R2);  CALL SYMPUT('L80RND',R3);
CALL SYMPUT('ALPHAN',ALPHAN);  CALL SYMPUT('BETA',BETA);
OK2:  CALL SYMPUT('DNE',DNE);
* PUT 'ALPHA BETA L50 AND NEXT DESIGN POINTS ARE 'ALPHA
  BETA XSMF50 XSMF20 XSMF80;
*;
*;
 GOPTIONS DEVICE = HERCULES;
DATA BOX;
  LENGTH FUNCTION $ 8.; XSYS='2'; YSYS='2';
  FUNCTION = 'MOVE'; X=&L20; Y=0; OUTPUT;
  FUNCTION = 'DRAW'; X=&L20; Y=.2; COLOR='BLUE'; LINE=20;
    OUTPUT;
  FUNCTION = 'MOVE'; X=&LG2; Y=.2; OUTPUT;
  FUNCTION = 'DRAW'; X=&L20; Y=.2; COLOR='BLUE'; LINE=20;
    OUTPUT;
  FUNCTION = 'MOVE'; X=&L80; Y=0; OUTPUT;
  FUNCTION = 'DRAW'; X=&L80; Y=.8; COLOR='BLUE'; LINE=20;
    OUTPUT;
  FUNCTION = 'MOVE'; X=&LG2; Y=.8; OUTPUT;
  FUNCTION = 'DRAW'; X=&L80; Y=.8; COLOR='BLUE'; LINE=20;
    OUTPUT;
  FUNCTION = 'MOVE'; X=(&L80 + &OFS); Y=.5; OUTPUT;
  FUNCTION = 'LABEL'; SIZE = 1.5; POSITION='6';
    TEXT = 'NEXT DESIGN POINTS:'; COLOR='BLUE'; OUTPUT;
  FUNCTION = 'MOVE'; X=(&L80 + &OFS); Y=.375; OUTPUT;
  FUNCTION = 'LABEL'; SIZE = 1.5; POSITION='6';
    TEXT = "&L20RND"; COLOR='BLUE'; OUTPUT;
  FUNCTION = 'MOVE'; X=(&L80 + &OFS); Y=.25; OUTPUT;
  FUNCTION = 'LABEL'; SIZE = 1.5; POSITION='6';
    TEXT = "&L80RND"; COLOR='BLUE'; OUTPUT;
  FUNCTION = 'MOVE'; X=(&L50 - &OFS); Y=.7; OUTPUT;
  FUNCTION = 'LABEL'; SIZE=1.5; POSITION = '4';
    TEXT = 'LD50 ESTIMATE'; COLOR='BLUE'; OUTPUT;
  FUNCTION = 'MOVE'; X=(&L50 - &OFS); Y=.575; OUTPUT;
  FUNCTION = 'LABEL'; SIZE = 1.5; POSITION = '4';
    TEXT = "&L50RND"; COLOR='BLUE'; OUTPUT;
DATA POINTS;
```

```
  DO X=&LG2 TO &UG2 BY &BYN; LOGIT=1/(1 +
                      EXP(-&BETA*(X-&ALPHAN)));
  OUTPUT; END;
PROC GPLOT DATA=POINTS;
  TITLE1 H=2.75 F=ITALIC J=C U=1 'SAM';
  AXIS1 LABEL = (H=.2 F=DUPLEX '')
        ORDER=0 TO 1 BY .2 VALUE=(T=1 H=.1 ' ' H=1.7 T=2
              '2' T=3 H=.1 ' ' T=4 H=.1 ' ' H=1.7 T=5 '.8'
              T=6 H=.1 ' ');
  AXIS2 LABEL = (H=1 f=DUPLEX 'X ');
  SYMBOL C=R I=JOIN V=NONE;
  PLOT LOGIT*X / VAXIS=AXIS1 HAXIS=AXIS2 ANNOTATE=BOX;
RUN; QUIT;



*                         SAMSUMM                        *;

**********************************************************;
***          ENTER THE FOLLOWING INFORMATION         ****;
***                                                  ****;
* ENTER THE NAME OF THE DATASET CONTAINING THE           *;
* OBSERVATIONS IN QUOTES (INCLUDE THE PATH AND FILENAME  *;
*                    IE. 'A:\SAMDAT.DAT')                *;
              FILENAME DAT1 'A:TEST3.DAT';
* ENTER THE NAME OF THE OUTPUT FILE (IT WILL CONTAIN THE *;
* SUMMARY STATSTICS) USE THE SAME FORMAT AS ABOVE        *;
              FILENAME SUMOUT 'A:SUMOUT';
*                                                        *;
**********************************************************;
***         NO CHANGES NEEDED BEYOND THIS POINT       ***;
**********************************************************;
*                                                        *;
DATA DAT; INFILE DAT1;
  KEEP X Y;
  INPUT OBS X Y;
PROC MEANS NOPRINT DATA=DAT N MAX MIN;
  VAR X; OUTPUT OUT=STATS N=NUM MAX=MX MIN=MN;
DATA STATS; SET STATS;
  LENGTH NUMC $ 8; NUMC = NUM; NUMC = LEFT(NUMC);
  NUMX = 'X'||TRIM(NUMC); NUMY = 'Y'||TRIM(NUMC);
  CALL SYMPUT('NUMX',NUMX); CALL SYMPUT('NUMY',NUMY);
  CALL SYMPUT('MX',MX); CALL SYMPUT('MN',MN);
  CALL SYMPUT('NUM',NUM);
PROC TRANSPOSE DATA=DAT OUT=TRANS;
  VAR X Y;
DATA DATX; SET TRANS;
  ARRAY X {&NUM} X1-&NUMX; KEEP X1-&NUMX DUM;
  IF _NAME_ = 'X'; DUM=1;
  X1=COL1; X2=COL2; X3=COL3; X4=COL4; X5=COL5; X6=COL6;
  X7=COL7; X8=COL8; X9=COL9; X10=COL10; X11=COL11;
  X12=COL12;  X13=COL13; X14=COL14; X15=COL15; X16=COL16;
  X17=COL17; X18=COL18; X19=COL19; X20=COL20; X21=COL21;
```

```
  X22=COL22; X23=COL23; X24=COL24;
  IF &NUM GT 20 THEN DO;
    X21=COL21; X22=COL22; X23=COL23; X24=COL24; X25=COL25;
    X26=COL26; X27=COL27; X28=COL28; X29=COL29; X30=COL30;
  END;
  IF &NUM GT 30 THEN DO;
    X31=COL31; X32=COL32; X33=COL33; X34=COL34; X35=COL35;
    X36=COL36; X37=COL37; X38=COL38; X39=COL39; X40=COL40;
  END;
  IF &NUM GT 40 THEN DO;
    X41=COL41; X42=COL42; X43=COL43; X44=COL44; X45=COL45;
    X46=COL46; X47=COL47; X48=COL48; X49=COL49; X50=COL50;
  END;
  IF &NUM GT 50 THEN DO;
    X51=COL51; X52=COL52; X53=COL53; X54=COL54; X55=COL55;
    X56=COL56; X57=COL57; X58=COL58; X59=COL59; X60=COL60;
  END;
DATA DATY; SET TRANS;
  ARRAY Y {&NUM} Y1-&NUMY; KEEP Y1-&NUMY DUM;
  IF _NAME_ = 'Y'; DUM=1;
  Y1=COL1; Y2=COL2; Y3=COL3; Y4=COL4; Y5=COL5; Y6=COL6;
  Y7=COL7; Y8=COL8; Y9=COL9; Y10=COL10; Y11=COL11;
  Y12=COL12; Y13=COL13; Y14=COL14; Y15=COL15; Y16=COL16;
  Y17=COL17; Y18=COL18; Y19=COL19; Y20=COL20;
  IF &NUM GT 20 THEN DO;
    Y21=COL21; Y22=COL22; Y23=COL23; Y24=COL24; Y25=COL25;
    Y26=COL26; Y27=COL27; Y28=COL28; Y29=COL29; Y30=COL30;
  END;
  IF &NUM GT 30 THEN DO;
    Y31=COL31; Y32=COL32; Y33=COL33; Y34=COL34; Y35=COL35;
    Y36=COL36; Y37=COL37; Y38=COL38; Y39=COL39; Y40=COL40;
  END;
  IF &NUM GT 40 THEN DO;
    Y41=COL41; Y42=COL42; Y43=COL43; Y44=COL44; Y45=COL45;
    Y46=COL46; Y47=COL47; Y48=COL48; Y49=COL49; Y50=COL50;
  END;
  IF &NUM GT 50 THEN DO;
    Y51=COL51; Y52=COL52; Y53=COL53; Y54=COL54; Y55=COL55;
    Y56=COL56; Y57=COL57; Y58=COL58; Y59=COL59; Y60=COL60;
  END;
DATA DAT; MERGE DATX DATY;
  BY DUM; DROP DUM;
PROC PRINT DATA=DAT; TITLE 'TRANSPOSED DATASET';
***                     RESCALE THE DATA
***;
DATA DAT; SET DAT;
  ARRAY X {&NUM} X1-&NUMX; ARRAY Y {&NUM} Y1-&NUMY;
  RANG = &MX - &MN; MID = (&MX + &MN) / 2;
  DO I = 1 TO &NUM;
    X{I} = (X{I} - MID)*(8/RANG);
  END;
PROC PRINT DATA=DAT; TITLE 'RESCALED DATASET';
DATA ONE; SET DAT;
```

```
***********************************************************;
***                INITIALIZE VARIABLES
***********************************************************;
  N = &NUM; CHK=0; DNE=0;
FILE PRINT;
  ARRAY X {&NUM} X1-&NUMX; ARRAY Y {&NUM} Y1-&NUMY;
    ALPHA = .25; BETA = .5;
***********************************************************;
***            CHECK SILVAPULLES CONDITIONS
***********************************************************;
    X0MIN = 50; X1MIN = 50;
    X0MAX = -50; X1MAX = -50;
    DO I = 1 TO N;
      IF Y{I} = 0 THEN DO;
        IF X{I} GT X0MAX THEN X0MAX = X{I};
        IF X{I} LT X0MIN THEN X0MIN = X{I}; END;
      IF Y{I} = 1 THEN DO;
        IF X{I} GT X1MAX THEN X1MAX = X{I};
        IF X{I} LT X1MIN THEN X1MIN = X{I}; END;
    END;
***                CONDITION 1                           ***;
    IF X1MAX GT X1MIN AND X0MIN LT X0MAX AND X1MAX GT X0MIN
AND
       X1MIN LT X0MAX THEN CHK = 1;
***                CONDITION 2                           ***;
    IF X0MIN = X0MAX AND X1MIN LT X0MIN AND X0MAX LT X1MAX
       THEN CHK = 1;
***                CONDITION 3                           ***;
    IF X1MIN = X1MAX AND X0MIN LT X1MIN AND X1MAX LT X0MAX
       THEN CHK = 1;
    IF CHK = 1 THEN GO TO SP1;
      PUT 'MLE DOES NOT EXIST'; DNE=1;
      CALL SYMPUT('DNE',DNE); GO TO OK;
***********************************************************;
***                  CALCULATE MLE'S
***********************************************************;
SP1:    G1=0; G2=0;
    DO J = 1 TO 10;
      IF (G1**2 + G2**2) LT .0001 AND J NE 1 THEN GO TO OK1;
      H11 = 0; H12 = 0; H22 = 0;
      G1 = 0; G2 = 0;
      DO I = 1 TO N;
        Z = ALPHA + BETA*X{I};
        IF ABS(Z) GT 15 THEN DO; PRED=1; GO TO DE3; END;
        PRED = EXP(Z) / (1 + EXP(Z));
DE3:    H11 = H11 - PRED*(1-PRED);
        H12 = H12 - X{I}*PRED*(1-PRED);
        H22 = H22 - X{I}*X{I}*PRED*(1-PRED);
        G1 = G1 + (Y{I} - PRED);
        G2 = G2 + (X{I}*Y{I} - X{I}*PRED);
      END;
      IF (H11*H22) - (H12**2) LT .001 THEN DO;
       PUT 'DET NEAR ZERO'; DNE=1; GO TO OK; END;
      DET = (H11*H22) - (H12**2);
```

```
      HINV11 = H22 / DET;
      HINV22 = H11 / DET;
      HINV12 = -(H12 / DET);
      ALPHA = ALPHA - ((HINV11*G1) + (HINV12*G2));
      BETA  = BETA  - ((HINV12*G1) + (HINV22*G2));
    END;
OK1:  FILE PRINT;
***         RESCALE DATA AND ESTIMATES TO NORMAL        ***;
    BETA = BETA*8/(&MX - &MN); ALPHA = (-ALPHA/BETA) +
          ((&MX + &MN)/2);
    ALPHA = -ALPHA*BETA;
    DO V=1 TO &NUM;
      X{V} = X{V}*((&MX - &MN)/8) + ((&MX + &MN) / 2);
    END;
    ALPHAN = -ALPHA/BETA; CALL SYMPUT('ALPHAN',ALPHAN);
    LG2 = ALPHAN - (LOG(75)/BETA); UG2 = ALPHAN +
          (LOG(75)/BETA);
    BYN = (UG2-LG2)/300; OFS = (UG2-LG2)/75;
    CALL SYMPUT('BYN',BYN); CALL SYMPUT('OFS',OFS);
    CALL SYMPUT('ALPHA',ALPHA); CALL SYMPUT('BETA',BETA);
    CALL SYMPUT('LG2',LG2); CALL SYMPUT('UG2',UG2);
    SYMPUT('BETAR',BETAR);
OK:  CALL SYMPUT('DNE',DNE);
*;
*;
DATA TWO; SET ONE;
  ARRAY X {&NUM} X1-&NUMX; ARRAY Y {&NUM} Y1-&NUMY;
  LENGTH ALPHAC $ 10 BETAC $ 10;
  ALPHA = ROUND(&ALPHA,.001); BETA = ROUND(&BETA,.001);
FILE PRINT; PUT 'ALPHA AND BETA ARE ' ALPHA BETA;
  FILE SUMOUT; NUM = &NUM;
  ALPHAC = ALPHA; BETAC = BETA;
  ALPHAC = LEFT(ALPHAC); BETAC = LEFT(BETAC);
****     PRINT DATASET      *****;
  PUT @30 'DESIGN POINTS' @48 'RESPONSE' OVERPRINT
      @30 '_____________' @48 '________';
  PUT ;
  DO J=1 TO NUM;
   PUT @32 X{J} 6.2  @50 Y{J} 2.;
  END;
****     CALCULATE ESTIMATES AND STANDARD ERRORS     *****;
  SUM1=0; SUM2=0;
  DO I=1 TO NUM;
    PI = (1 + EXP(-&BETA*(X{I} - &ALPHAN)))**(-1);
    SUM1 = SUM1 + PI*(1 - PI);
    SUM2 = SUM2 + (X{I}**2)*PI*(1 - PI);
  END;
  SEALPHA = SQRT(1 / (n*SUM1)); SEBETA = SQRT(1 / (n*SUM2));
  PUT ' ';
  PUT @30 'ALPHA = ' ALPHA  6.3 '  BETA = ' BETA 6.3;
  PUT @30 'WITH STANDARD ERRORS OF ' SEALPHA 8.3 SEBETA 8.3;
  PUT ' ';
  PUT @16 'PERCENTILE' @30 'ESTIMATE' @40 'S.E.' @50 'LOWER
      95%' @60 'UPPER 95%' OVERPRINT @16 '__________' @30
```

```
      '________'  @40 '____' @50 '_________'
      @60 '_________'; PUT ' ';
  DO PS = .1 TO .9 BY .1;
    LPS = &ALPHAN - (LOG((1-PS)/PS)/&BETA);
    C = LOG(PS/(1-PS)) / LOG(1/4);
    TERM1 = 1 / (.16*&NUM*(&BETA**2));
    VARLPS = (TERM1/4)*((1+C)**2 + (1-C)**2);
    SELPS = SQRT(VARLPS);
* BUILD MINI T-TABLE TO CALCULATE THE APPROPRIATE T-VALUE *;
     IF NUM LE 8 THEN TTAB = 2.5;
     IF NUM GT 8 AND NUM LE 11 THEN TTAB = 2.228;
     IF NUM GT 11 AND NUM LE 14 THEN TTAB = 2.145;
     IF NUM GT 14 AND NUM LE 17 THEN TTAB = 2.120;
     IF NUM GT 17 AND NUM LE 20 THEN TTAB = 2.093;
     IF NUM GT 20 AND NUM LE 25 THEN TTAB = 2.069;
     IF NUM GT 25 AND NUM LE 30 THEN TTAB = 2.048;
     IF NUM GT 30 AND NUM LE 40 THEN TTAB = 2.031;
     IF NUM GT 40 AND NUM LE 60 THEN TTAB = 2.010;
     IF NUM GT 60 AND NUM LE 120 THEN TTAB = 1.99;
     IF NUM GT 120 THEN TTAB = 1.96;
* END MINI T-TABLE (t .025 one sided values)              *;
    LCL = LPS - TTAB*SELPS; UCL = LPS + TTAB*SELPS;
    IF LCL LT 0 THEN LCL=0;
    PUT @18 PS  @30 LPS 6.2  @38 SELPS 6.2 @50 LCL 6.2
    @60 UCL 6.2; PUT ;
  END;
L80 = &ALPHAN + (LOG(4)/&BETA); L50RND = ROUND(&ALPHAN,.01);
CALL SYMPUT('ALPHAC',ALPHAC); CALL SYMPUT('BETAC',BETAC);
CALL SYMPUT('L80',L80); CALL SYMPUT('L50RND',L50RND);
*;
*;
GOPTIONS DEVICE = HERCULES;
DATA BOX;
  LENGTH FUNCTION $ 8.; XSYS='2'; YSYS='2';
  FUNCTION = 'MOVE'; X=&ALPHAN; Y=0; OUTPUT;
  FUNCTION = 'DRAW'; X=&ALPHAN; Y=.5; COLOR='BLUE'; LINE=20;
    OUTPUT;
  FUNCTION = 'MOVE'; X=&LG2; Y=.5; OUTPUT;
  FUNCTION = 'DRAW'; X=&ALPHAN; Y=.5; COLOR='BLUE'; LINE=20;
    OUTPUT;
  FUNCTION = 'MOVE'; X=(&L80 + &OFS); Y=.5; OUTPUT;
  FUNCTION = 'LABEL'; SIZE = 1.5; POSITION='6';
    TEXT = 'FINAL ESTIMATES:'; COLOR='BLUE'; OUTPUT;
  FUNCTION = 'MOVE'; X=(&L80 + &OFS); Y=.375; OUTPUT;
  FUNCTION = 'LABEL'; SIZE = 1.5; POSITION='6';
    TEXT = "ALPHA = &ALPHAC"; COLOR='BLUE'; OUTPUT;
  FUNCTION = 'MOVE'; X=(&L80 + &OFS); Y=.25; OUTPUT;
  FUNCTION = 'LABEL'; SIZE = 1.5; POSITION='6';
   TEXT = "BETA = &BETAC"; COLOR='BLUE'; OUTPUT;
  FUNCTION = 'MOVE'; X=(&ALPHAN - &OFS); Y=.7; OUTPUT;
  FUNCTION = 'LABEL'; SIZE = 1.5; POSITION = '4';
   TEXT = 'LD50 ESTIMATE'; COLOR='BLUE'; OUTPUT;
  FUNCTION = 'MOVE'; X=(&ALPHAN - &OFS); Y=.575; OUTPUT;
  FUNCTION = 'LABEL'; SIZE = 1.5; POSITION = '4';
```

```
   TEXT = "&L50RND"; COLOR='BLUE'; OUTPUT;
DATA POINTS;
  DO X=&LG2 TO &UG2 BY &BYN; LOGIT = 1/(1 +
       EXP(-&BETA*(X-&ALPHAN))); OUTPUT;
  END;
PROC GPLOT DATA=POINTS;
  TITLE1 H=2.5 F=ITALIC J=C U=1 'SAM';
  AXIS LABEL = (H=.2 F=DUPLEX ' ')
       ORDER=0 TO 1 BY .2 VALUE= (T=1 H=.1 ' ' H=1.7 T=2 '.2'
               T=3 H=.1 ' ' T=4 H=.1 ' ' H=1.7 T=5 '.8' T=6
               H=.1 ' ');
  AXIS2 LABEL = (H=1 F=DUPLEX 'X ');
  SYMBOL C=R I=JOIN V=NONE;
  PLOT LOGIT*X / VAXIS=AXIS1 HAXIS=AXIS2 ANNOTATE=BOX;
RUN;QUIT;
```

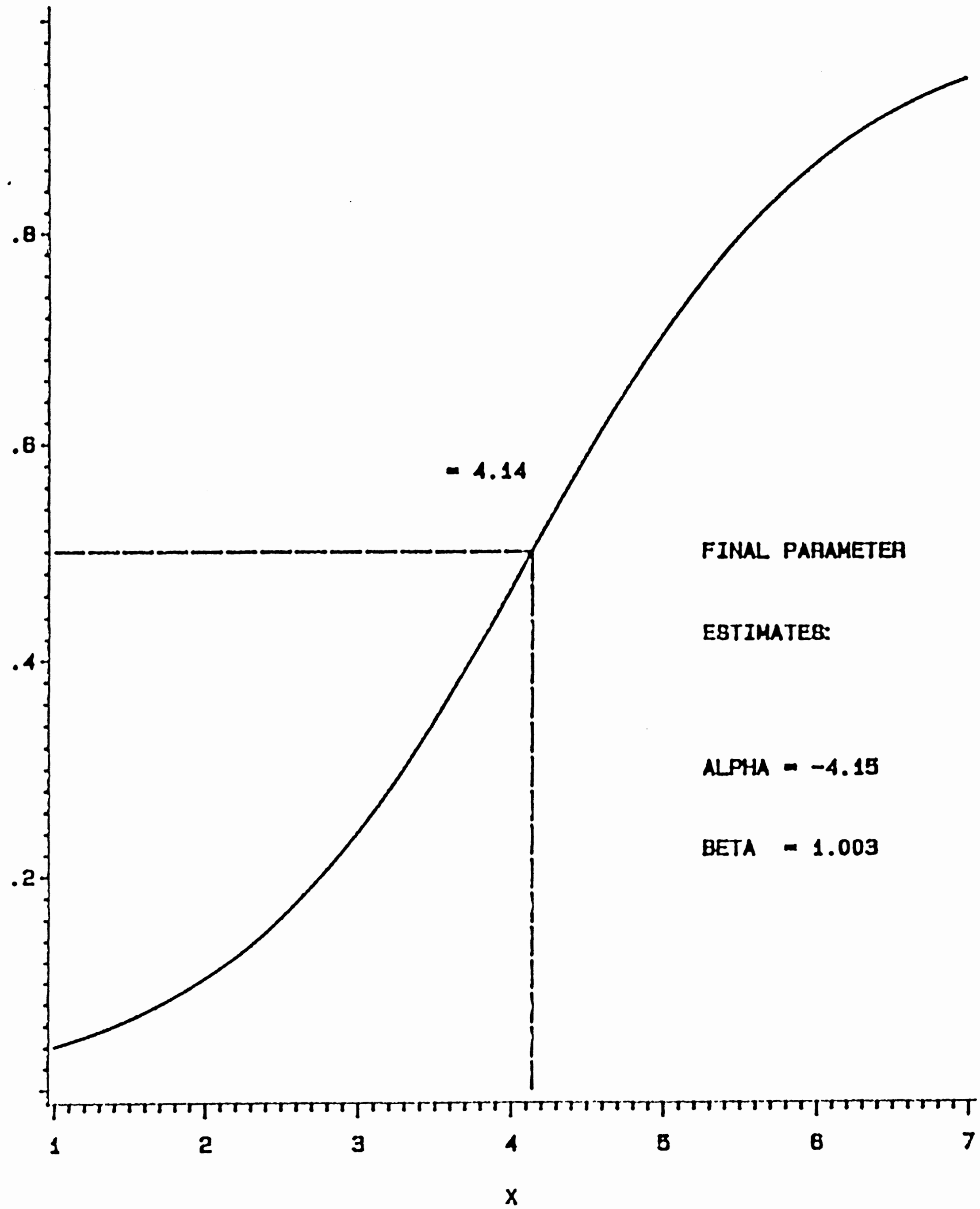

Figure 6. SAMSUMM Output

DESIGN POINTS	RESPONSE
2.00	0
4.00	0
2.00	0
4.50	1
3.00	0
4.75	0
3.00	1
5.00	1
4.00	0
5.00	1
3.04	1
5.46	1
2.35	0
5.32	1
2.62	0
5.06	1
2.79	0
4.91	0
2.92	0
5.33	1
3.04	1
5.20	0
2.60	0
5.70	1

ALPHA = -4.150   BETA = 1.003

PERCENTILE	ESTIMATE	S.E.	LOWER 95%	UPPER 95%
10	1.95	0.67	0.55	3.34
20	2.76	0.51	1.70	3.81
30	3.29	0.42	2.42	4.16
40	3.73	0.37	2.96	4.51
50	4.14	0.36	3.39	4.88
60	4.54	0.37	3.77	5.32
70	4.98	0.42	4.11	5.85
80	5.52	0.51	4.47	6.57
90	6.33	0.67	4.93	7.72

Figure 6 (continued)

VITA

Douglas E. Faries

Candidate for the degree of

Doctor of Philosophy

Thesis: A NEW SEQUENTIAL APPROXIMATION METHOD

Major Field: Statistics

Biographical:

Personal Data: Born in Murray, Kentucky, September, 11, 1963, to Clyde and Elizabeth Faries. Married Karen Stanton, of Russellville, Arkansas, on December 17, 1988.

Education: Graduated from Macomb High School, Macomb, Illinois, in June, 1981; Received a Bachelor of Science degree (December, 1984) and a Master of Science degree (August, 1986) in Mathematics from Western Illinois University; Completed the requirements for a Doctor of Philosophy degree in Statistics at Oklahoma State University in May, 1990.

Professional Experience: Teaching Assistant (Elementary and Intermediate Algebra), Department of Mathematics, Western Illinois University, January, 1985, to May, 1986; Research Assistant, Department of Mathematics, Western Illinois University, June, 1986, to August, 1986; Research Assistant to Dr. Ron McNew (statistical consultant to the Agricultural Experiment Station), Oklahoma State University, August, 1986, to May, 1988; Summer Intern Statistician, Eli Lilly & Co. (Pre-Clinical), Greenfield, Indiana, May, 1988, to July, 1988; Teaching Assistant (Statistical Methods I, Engineering Statistics), Oklahoma State University, Department of Statistics, August, 1988, to May, 1990; Summer Intern Statistician, E.I. DuPont deNemours & Co. (Biometrics Department), Wilmington, Delaware, May, 1989, to August, 1989.

Scholarly Organizations: American Statistical Association member, OSU Statistics Club (President, 1988), WIU Math Club (Vice President, 1985), Phi Kappa Phi

Scholarships and Honors: Robert Glenn Rapp Distinguished Graduate Fellowship (OSU), Oklahoma State Foundation Scholarship, Olive and Grove Billingsley Tyson Scholarship (WIU), Western Illinois University Foundation Scholarship, Jacob Burlak Award (WIU Mathematics Department)